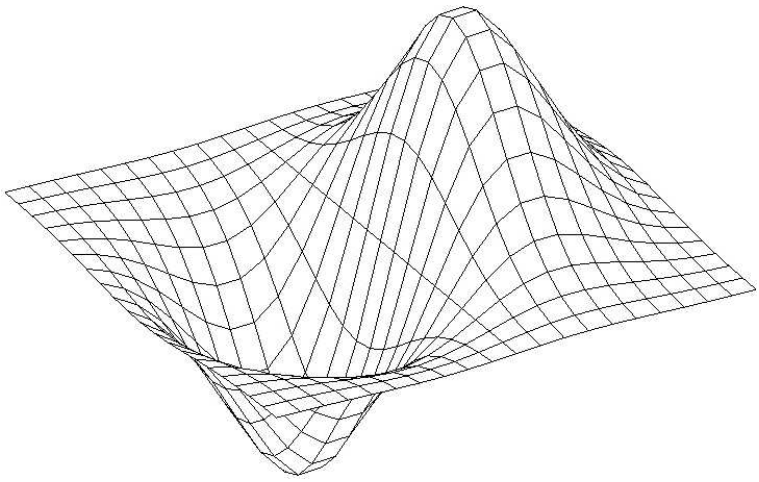
Institute of Geonics AS CR, Ostrava

SNA'13

Seminar on Numerical Analysis

Modelling and Simulation
of Challenging Engineering Problems



Winter School

Methods of Numerical Mathematics and Modelling,
High-Performance Computing, Numerical Linear Algebra

Rožnov pod Radhoštěm, January 21 – 25, 2013

Programme committee:

Radim Blaheta	Institute of Geonics AS CR, Ostrava
Zdeněk Dostál	VŠB-Technical University, Ostrava
Ivo Marek	Czech Technical University, Prague
Miroslav Rozložník	Institute of Computer Science AS CR, Prague
Zdeněk Strakoš	Charles University, Prague

Organizing committee:

Hana Bílková	Institute of Computer Science AS CR, Prague
Radim Blaheta	Institute of Geonics AS CR, Ostrava
Eva Dudková	Institute of Geonics AS CR, Ostrava
Jiří Starý	Institute of Geonics AS CR, Ostrava

Conference secretary:

Jaroslava Vávrová	Institute of Geonics AS CR, Ostrava

Institute of Geonics AS CR
ISBN 978-80-86407-34-0

Preface

Seminar on Numerical Analysis 2013 (SNA'13) is the tenth meeting in a series of events which started ten years ago in Ostrava 2003. The following meetings were held in Ostrava 2005, Moninec 2006, Ostrava 2007, Liberec 2008, Ostrava 2009, Nové Hrady 2010, Rožnov 2011, Liberec 2012. The tenth SNA 2013 is again held in Rožnov pod Radhoštěm at the hotel Relax.

Since 2005, a part of SNA is devoted to Winter School with tutorial lectures devoted to selected important topics within the scope of numerical methods and modelling. This year, this part includes invited lectures devoted to adaptivity for linear and nonlinear solvers (Vohralík), stochastic finite elements (Sousedík), algebraic multigrid, stochastic problems and homogenization (Marek, Pultarová), fast solvers and parallelism in the boundary element method (Lukáš), multigrid methods for problems of mathematical physics and multiphysics (Hron).

Like the first SNA 2003, the present SNA 2013 is also devoted to the jubilee of Prof. RNDr. Ivo Marek, DrSc., our colleague, teacher, friend who strongly influenced the development of numerical analysis in our country. In many respects, SNA conference series is a follower of series Software and Algorithm of Numerical Analysis (SANM) with Ivo Marek as the main organizer. The SANM conferences were held for thirty years, being organized each second year, starting in 1975.

Looking back at least over the ten years history of SNA, it is a pleasure to see that many participants grown from students to recognized distinguished scientists, that there are new young students and colleagues interested in the numerical analysis and that some ideas remain valid for all times. To support the last statement, let us remember that in the announcement of SNA 2003 we mentioned an anonymous general principle

the faster the computer, the more
important the speed of algorithms

which, we are convinced is valid and maybe even more important nowadays, when we start the first supercomputing project IT4Innovations in the Czech Republic.

Besides IT4Innovations, it is also our pleasure to acknowledge the support from the project SPOMECH "Creation of Multidisciplinary Team for Reliable Solution of Nonlinear Problems of Mechanics", reg. no. CZ.1.07/2.3.00/20.0070.

Let us wish SNA 2013 to be, similarly to the previous SNA meetings, a fruitful event, providing interesting lectures, showing new ideas, beauty of numerical analysis and starting or strengthening collaboration and friendship.

On behalf of the Programme and Organizing Committee of SNA 2013,

Radim Blaheta and Jiří Starý

Laudatio on Prof. RNDr. Ivo Marek, DrSc.

Ivo Marek was born on the 24th January 1933 in Prague. After finishing classical gymnasium, he decided to study mathematics at the Charles University in Prague and graduated here in 1956 with diploma thesis supervised by a famous Czech mathematician Vojtěch Jarník. His thesis was devoted to the number theory, especially to grid numbers. Both grids and numbers can be found in his later work, although in a rather different context.

After graduating from university, Ivo Marek was by an administrative decision sent to work as a computational mathematician in the Nuclear Research Institute at Řež near Prague, at that time a new and rapidly developing institution. Such administrative decision was common at that time and considering his case, it was very lucky. He came there in contact with problems of reactor physics, which influenced his later scientific work and gave him a lively interest in deep applications of mathematics in physics and engineering. Ivo started from analytical solution but soon became interested in functional analysis, theory of operators and analysis of deep and important problems.

The hard scientific work, which started in Řež, resulted in obtaining the scientific degrees CSc. (1962 - Iteration of Nonlinear Bounded Operators and Iterative Processes in Nonselfadjoint Eigenvalue Problems) and DrSc. (1968), habilitation (1965) and getting a new job at the Mathematical Institute of the Charles University (from 1963). His scientific development was admirably fast, e.g. his CSc. (PhD.) thesis was prepared and defended in one year! On the other hand, he also managed to play tennis, and even more, with his wife they became winners of the regional tennis league!

In 1967 Ivo Marek visited Novosibirsk and met here a number of distinguished scientists. Let us mention primarily G.I. Marchuk and G.E. Forsythe, who later invited him to the USA. This led to him getting a position of a visiting professor at Case Western Reserve University, Cleveland, Ohio (1968 - 1970) and University of Wisconsin (1970). Here Ivo Marek met a lot of other famous mathematicians; we can mention such names as Varga, Householder, Wilkinson, Fox, Golub, Nickel, Aubin and Schneider. Many of them shared his enthusiasm for math and tennis. At several conferences he played tennis matches also with G. Strang and I can imagine that the topic of my first mathematical work supervised by Ivo Marek could have its origin just there. This period was very fruitful for the numerical analysis in many respects. Ivo's host, R. Varga then wrote a beautiful book "Matrix iterative analysis" and continued to work in iterative methods. By a lucky chance, Ivo could inform R. Varga and Ph. Ciarlet about a new pioneering paper by M. Zlámal "On the finite element method".

For us it was important that after returning home in 1970, Ivo Marek was appointed as the head of the Department of Numerical Mathematics at the Faculty of Mathematics and Physics of the Charles University. Here he exploited much from his experience. He and his colleagues introduced a lot of new courses based on functional analysis, modern theory of partial differential equations and new achievements in numerical methods. The courses referred to distinguished textbooks and monographs by A. Ralston, A. Taylor, J. Ortega and W. Rheinboldt, R. Varga, J. Fix and G. Strang etc. Ivo gave the courses on theory of matrices, which later resulted in a two-volume monograph "Theory of Matrices in Applied Sciences" written with K. Žitný. In 1977 Ivo Marek was appointed the full professor of mathematics at the Charles University.

Besides the basic knowledge, Prof. Marek transferred to students his enthusiasm for mathematics, for finding hidden relations and seeking new points of view. He transferred to us also the feeling of the worldwide dimension of the science, which was especially important in the seventies, when our society felt somewhat isolated from a part of the world.

Ivo was also active as an organizer of many seminars and conferences. Let us mention here the successful series of Summer Schools on Software and Algorithms of Numerical Mathematics, which started in 1975 at Zadov and proceeded each second year for thirty years. These conferences were very important for development of the numerical analysis and application of the numerical methods in our country. From the other conferences, we should not omit the international conferences ISNA 1985, 1987, 1990, 1992, which were held alternatively in Prague and Madrid. At that time Ivo was also appointed an Honorary Professor of the Universidad Politecnica de Madrid, Spain. Later, Ivo participated in organization of the annual GAMM Conference in Prague in 1995; he was also involved in organizing of two important conferences on Computational Linear Algebra at Milovy 1997 and 2002. From 2003 he is a member of the SNA Programme Committee.

From 1996, Ivo Marek also started to teach at the Czech Technical University in Prague and found a new space for application of his broad knowledge here. He found new colleagues and students and contributed to their research in the field of engineering. But especially, he made a great progress and obtained new excellent results in application of iterative methods for solving problems with stochastic matrices. Many new results on this topics can be found in papers with several coworkers, Daniel Szyld from Temple University, Petr Mayer and Ivana Pultarová from CTU Prague. We are glad, that we can be further acquainted with these results at the Winter School lectures at this SNA. The list of his research interest is definitely much broader. I personally, together with the whole group of mathematicians from Ostrava, would like to appreciate very much his interest and encouragement, which helps us in many cases.

The worldwide scientific reputation of Ivo Marek resulted in his membership in editorial boards of several scientific journals; the most prestigious of them are Numerical Linear Algebra with Applications, Numerical Functional Analysis, and Numerical Methods for Partial Differential Equations, Integral Transforms and Special Functions. His work was awarded e.g. by the National Price and B. Bolzano Medal for Merits in the Mathematical Sciences (Czechoslovak Academy of Sciences).

Ivo's enthusiasm for numerical linear algebra, functional analysis and mathematics in general, and unfailing friendliness have brought him many friends all over the world. It deeply impressed me, when in nineties I got my first opportunities to accompany him to conferences abroad. The discussions with other participants usually showed the width and depth of Ivo Marek's mathematical knowledge and interests and, usually, he was also a centre of the fun and an excellent companion.

After finishing this brief and incomplete enlightenment of Ivo's exceptional personality, I would like personally and on behalf of the conference participants to wish Ivo good health, happiness and many further successes in his activities. We wish him to always be an optimist with unbounded energy and a source of enthusiasm surrounded by friends, colleagues and students.

Radim Blaheta

SPOMECH project

The main goal of the SPOMECH project (European Regional Development Fund, reg. no. CZ.1.07/2.3.00/20.0070) is to create a multidisciplinary research team working in the field of reliable modelling of nonlinear problems of mechanics and geomechanics and promote research activities and international cooperation in these subjects.

The project also supports seminars, invitations of specialists, three international workshops and a final conference, all in the period from July 2011 to June 2014. Besides SNA, the main SPOMECH supported events include:

- **1st SPOMECH Workshop**, Ostrava, November 22 - 24, 2011

  Main speakers: Wolfgang Hackbusch (Leipzig), Sergey Repin (St. Petersburg), Johannes Kraus (Linz)

- **2nd SPOMECH Workshop**, Ostrava, November 19 - 20, 2012

  Main speakers: Maya G. Neytcheva (Uppsala University), Talal Rahman (Bergen University), Alexander Popp (Technical University Munich), François-Xavier Roux (ONERA), Frédéric Feyel (ONERA)

We can also mention the planned future events:

- **3rd SPOMECH Workshop**, Ostrava, November 2013
- **Seminar on Numerical Analysis SNA 2014**
- **MODELLING 2014 conference**, June 2014

  The scope is computational modelling in engineering and science: multiscale modelling, multiphysics modelling, progress in discretization methods, efficient solvers, nonlinear problems, challenging applications of mathematical modelling methods in engineering.

There are also events organized in strong collaboration with the SPOMECH team:

- **Autumn School on Parallel Solution of Large Engineering Problems**
  Ostrava, November 19 - 23, 2012

  Main speakers: Johannes Kraus (RICAM), Svetozar Margenov (BAS Sofia), Oliver Rheinbach (University of Duisburg-Essen), Erhan Turan (ETH Zurich), Roman Wyrzykowski (Czestochova University of Technology)

- **High Performance Computing in Science and Engineering HPCSE 2013**
  Hotel Soláň, Beskydy Mountains, CR, May 27 - 30, 2013

- **Preconditioning of Iterative Methods: Theory and Applications PIM 2013**
  Prague, July 1 - 5, 2013

evropský sociální fond v ČR
EVROPSKÁ UNIE
MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY
OP Vzdělávání pro konkurenceschopnost
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Contents

Winter school lectures

J. Hron
Multigrid methods for problems of mathematical physics and multiphysics

D. Lukáš
Efficient numerics for boundary integral equations

I. Marek, I. Pultarová
Algebraic multigrid, stochastic matrices and homogenization

B. Sousedík
Stochastic finite element methods

M. Vohralík
Adaptivity for linear and nonlinear solvers and time step and space mesh selection in numerical discretizations

Stopping criteria in the parallel one–sided block–Jacobi SVD algorithm

M. Bečka, G. Okša, M. Vajteršic

Institute of Mathematics SAS, Bratislava

The one-sided block-Jacobi SVD algorithm is suited for the SVD computation of a general complex matrix A of order $m \times n,\ m \geq n$. However, we will restrict ourselves to real matrices with obvious modifications for the complex case.

We start with the block-column partitioning of A in the form

$$A = [A_1, A_2, \ldots, A_\ell],$$

where the width of A_i is $n_i,\ 1 \leq i \leq \ell$, so that $n_1 + n_2 + \cdots + n_\ell = n$. Due to the computational balance and communication complexity in the case of parallel implementation, it is preferable to choose n_i of comparable size for all i.

The serial algorithm can be written as an iterative process:

$$\begin{aligned} A^{(0)} &= A, \quad V^{(0)} = I_n, \\ A^{(r+1)} &= A^{(r)}U^{(r)}, \quad V^{(r+1)} = V^{(r)}U^{(r)}, \quad r \geq 0. \end{aligned} \tag{1}$$

Here the $n \times n$ orthogonal matrix $U^{(r)}$ is the so-called *block rotation* of the form

$$U^{(r)} = \begin{pmatrix} I & & & & \\ & U_{ii}^{(r)} & & U_{ij}^{(r)} & \\ & & I & & \\ & U_{ji}^{(r)} & & U_{jj}^{(r)} & \\ & & & & I \end{pmatrix},$$

where the unidentified matrix blocks are zero. The purpose of matrix multiplication $A^{(r)}U^{(r)}$ in (1) is to mutually orthogonalize individual columns between block columns i and j of $A^{(r)}$. The matrix blocks $U_{ii}^{(r)}$ and $U_{jj}^{(r)}$ are square of order n_i and n_j, respectively, while the first, middle and last identity matrix is of order $\sum_{s=1}^{i-1} n_s$, $\sum_{s=i+1}^{j-1} n_s$ and $\sum_{s=j+1}^{r} n_s$, respectively. The orthogonal matrix

$$\hat{U}^{(r)} = \begin{pmatrix} U_{ii}^{(r)} & U_{ij}^{(r)} \\ U_{ji}^{(r)} & U_{jj}^{(r)} \end{pmatrix}$$

of order $n_i + n_j$ is called the *pivot submatrix* of $U^{(r)}$ at step r. During the iterative process (1), two index functions are defined: $i = i(r),\ j = j(r)$ whereby $1 \leq i < j \leq \ell$. At each step r, the pivot pair (i, j) is chosen according to a given *pivot strategy* that can be identified with a function $\mathcal{F}:\ \{0, 1, \ldots\} \to \mathbf{P}_\ell = \{(c, d) : 1 \leq c < d \leq \ell\}$. If $\mathbf{O} = \{(c_1, d_1), (c_2, d_2), \ldots, (c_{N(\ell)}, d_{N(\ell)})\}$ is some ordering of $\mathbf{P}_\ell$ with $N(\ell) = \ell\,(\ell - 1)/2$, then the *cyclic* strategy is defined by:

If $c \equiv \ell - 1 \mod N(\ell)$ then $(i(r), j(r)) = (c_s, d_s)$ for $1 \leq s \leq N(\ell)$.

The most common cyclic strategies are the *row-cyclic* one and the *column-cyclic* one, where the orderings are given row-wise and column-wise, respectively, with regard to the upper triangle of A. The first $N(\ell)$ iterations constitute the first *sweep*. When the first sweep is completed,

the pivot pairs (i, j) are repeated during the second sweep, and so on, up to the convergence of the entire algorithm.

Notice that in (1) only the matrix of right singular vectors $V^{(r)}$ is iteratively computed by orthogonal updates. If the process ends at iteration t, say, then $A^{(t)}$ has mutually highly orthogonal columns. Their norms are the singular values of A, and the normalized columns (with unit 2-norm) constitute the matrix of left singular vectors.

The parallel version of the one-sided block-Jacobi SVD algorithm implemented on p processors with the blocking factor $\ell = 2p$ is given in the form of Algorithm 1.

Algorithm 1 *Parallel one-sided block-Jacobi SVD algorithm*

$V = I_n,\ \ell = 2 * p$
▷ each processor has 2 block columns of A : A_L and A_R
$G = \begin{pmatrix} G_{LL} & G_{LR} \\ G_{LR}^T & G_{RR} \end{pmatrix} = \begin{pmatrix} A_L^T A_L & A_L^T A_R \\ A_R^T A_L & A_R^T A_R \end{pmatrix}$
▷ *global convergence criterion with a constant* ϵ, $0 < \epsilon \ll 1$
while $(F(A, \ell) \geq \epsilon)$ **do**
  ▷ *local convergence criterion with a constant* δ, $0 < \delta \ll 1$
  if $(F(G, \ell) \geq \delta)$ **then**
    ▷ *diagonalization of* G
    `EVD`(G, X)
    ▷ *update of block columns*
    $(A_L, A_R) = (A_L, A_R) * X$
    $(V_L, V_R) = (V_L, V_R) * X$
  end if
  ▷ *parallel ordering–choice of* p *independent pairs* (i, j) *of block columns*
  `ReOrderingComp`(p)
  `Send-Receive`$(A_s, V_s, \mathrm{diag}(G_{ss}))$, where s is either L or R
end while
sv_L : square roots of diagonal elements of G_{LL}
sv_R : square roots of diagonal elements of G_{RR}
▷ *two block columns of left singular vectors*
$U_L = A_L * \mathrm{diag}(1/sv_L),\ U_R = A_R * \mathrm{diag}(1/sv_R)$

end

Four variants of a new dynamic ordering were designed for the parallel one-sided block Jacobi SVD algorithm in [1]. Similarly to the two-sided algorithm, the dynamic ordering takes into account the actual status of a matrix—this time of its block columns with respect to their mutual orthogonality. Variants differ in the computational and communication complexities and in proposed global and local stopping criteria.

Variant 1 is based on a parallel implementation of the Lanczos processes applied to a set of the symmetric Jordan-Wielandt matrices C,

$$C \equiv \begin{pmatrix} 0 & A_i^T A_j \\ A_j^T A_i & 0 \end{pmatrix}.$$

The aim is to estimate the absolute values of L largest eigenvalues that are the cosines of L smallest principal angles between $\mathrm{span}(A_i)$ and $\mathrm{span}(A_j)$. Having p processors, p mutually most

inclined pairs (A_i, A_j) are chosen for orthogonalization at the beginning of each parallel iteration step. The mutual inclination is estimated by the weight

$$w_{ij}^{(1)} \equiv \|T_L\|_{\mathrm{F}}^2 = \sum_{s=1}^{L} \alpha_s^2 + 2 \sum_{s=2}^{L} \beta_s^2,$$

where the coefficients $\alpha_s = (Cz_s, z_s)$ and $\beta_{s+1} = \|Cz_s - \alpha_s z_s\|$ are elements of the symmetric, tri-diagonal matrix T_L. The global stopping criterion of the iteration process is based on the maximum value of currently computed weights $w_{ij}^{(1)}$. When using a computer with machine precision ϵ, the convergence is reached when

$$\max_{i,j} w_{ij}^{(1)} < m\, L\, \epsilon,$$

where m is the number of matrix rows and L is the number of steps in Lanczos processes. The local stopping criterion is similar: A given pair (i, j) of block columns is not mutually orthogonalized if

$$w_{ij}^{(1)} < m\, L\, \epsilon.$$

In variant 2, the mutual position of $\mathrm{span}(A_i)$ and $\mathrm{span}(A_j)$ is described by using just one representative vector per subspace,

$$c_i \equiv \frac{A_i\, e}{\|e\|},\ 1 \le i \le \ell,$$

where $e \equiv (1, 1, \ldots, 1)^T \in \mathbb{R}^{k \times 1}$. The weight is defined as

$$w_{ij}^{(2)} \equiv |(c_i, c_j)|,$$

and p largest weights define p pairs of block columns for the orthogonalization. The global stopping criterion takes into account that the computation of c_i requires no scalar product (only the sum of k columns of A_i), whereas to compute $w_{ij}^{(2)}$, one scalar product of length m is needed. In what follows, we *neglect* parameters m and k and take into account only the number of scalar products required for the computation of weights. However, for one scalar product we would directly work with the machine precision ϵ. Therefore, in this case, we define the global stop less strictly (and somewhat arbitrarily) as

$$\max_{i,j} w_{ij}^{(2)} < 10\, \epsilon.$$

With respect to local computation, two block columns are not mutually orthogonalized if

$$w_{ij}^{(2)} < 10\, \epsilon.$$

Variant 3 uses the weight

$$w_{ij}^{(3)} \equiv \|A_i^T c_j\| = \frac{\|A_i^T A_j e\|}{\|e\|},$$

where c_j is the representative vector for $\mathrm{span}(A_j)$. It can be shown that $w_{ij}^{(3)}$ is the locally optimal version of $w_{ij}^{(2)}$,

$$w_{ij}^{(3)} = \max_{\|y\|=1} |(A_i y, c_j)|$$

for given A_i and c_j. We have proposed the global stopping criterion for variant 3 as

$$\max_{i,j} w_{ij}^{(3)} < k\,\epsilon.$$

Locally, two block columns are not mutually orthogonalized if

$$w_{ij}^{(3)} < k\,\epsilon.$$

Finally, variant 4 computes the 'exact' weights

$$w_{ij}^{(4)} \equiv \|A_i^T A_j\|_{\mathrm{F}};$$

a small value of $w_{ij}^{(4)}$ means that span(A_i) is nearly orthogonal to span(A_j). Notice that this is not true for variant 2 because there is no lower bound for the value of $w_{ij}^{(2)}$—it can be nearly zero even if subspaces are significantly inclined to each other. The proposed global stopping criterion is

$$\max_{i,j} w_{ij}^{(4)} < k^2\,\epsilon,$$

and the corresponding local stopping criterion is of the form

$$w_{ij}^{(4)} < k^2\,\epsilon.$$

The performance of four variants of dynamic ordering was tested on square random matrices of order 4000 and 8000, with six different distributions of singular values and two condition numbers (10^1 for the well conditioned case and 10^8 for the ill conditioned one), using 16 and 32 processors. All variants of dynamic ordering were compared with two parallel cyclic orderings with respect to the number of parallel iteration steps needed for the convergence, total parallel execution time and relative error in the orthogonality of computed left singular vectors. It turns out that the variant 3, for which a local optimality in some precisely defined sense can be proved, is the most efficient one. Additional numerical experiments show that this recommended variant 3 is about 1.5 times faster than the parallel two-sided block–Jacobi algorithm with dynamic ordering, and about 2–3 times slower than the ScaLAPACK procedure `PDGESVD`.

Acknowledgments: Authors were supported by the VEGA grant no. no. 2/0003/11 from the Scientific Grant Agency of the Ministry of Education and Slovak Academy of Sciences, Slovakia.

References

[1] M. Bečka, G. Okša, M. Vajteršic: *Parallel one-sided block-Jacobi SVD algorithm with dynamic ordering*. Sent for publication in Parallel Computing, November 2012.

Tvarová optimalizace pro 2D kontaktní problém se zadaným třením s koeficientem tření závislým na řešení

P. Beremlijski, J. Haslinger, J. Outrata, R. Pathó

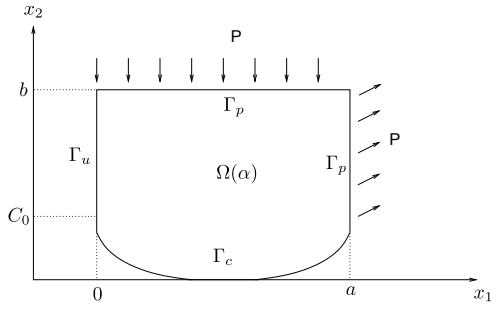
Centrum excelence IT4Innovations a Katedra aplikované matematiky
Vysoká škola báňská - Technická univerzita Ostrava

1 Úvod

V příspěvku se zabýváme diskretizovanou úlohou tvarové optimalizace dvojrozměrného pružného tělesa v jednostranném kontaktu s tuhou překážkou. Stavová úloha je v našem případě dána jako Signoriniho problém s Trescovým třením s koeficientem tření závislým na řešení. Při splnění jistých podmínek pro koeficient tření má diskrétní kontaktní úloha jediné řešení. Navíc řešení této úlohy je závislé lokálně lipschitovsky na řídící proměnné popisující tvar pružného tělesa. Díky jedinému řešení diskrétní úlohy pro fixovanou řídící proměnnou, můžeme použít tzv. přístup implicitního programování. Ten je založen na minimalizaci nehladké funkce složené z cenové funkce a jednoznačného zobrazení, které řídící proměnné přiřazuje řešení diskrétní úlohy, tzn. stavové proměnné. Pro minimalizaci nehladké funkce lze efektivně použít bundle trust metodu. K výpočtu subgradientní informace, kterou metoda vyžaduje, je nutné použít Morduchovičův kalkul. Na závěr příspěvku je ilustrováno použití našeho přístupu. Podrobně se lze s uvedeným přístupem seznámit v [3].

2 Stavová úloha

Nechť $\Omega \subset \mathbb{R}^2$ je pružné těleso s lipschitzovskou hranicí $\partial\Omega$. Hranice $\partial\Omega$ je složena ze tří nepřekrývajících se částí Γ_u, Γ_p a Γ_c. Viz obrázek 1.



Obrázek 1: 2D pružné těleso.

Γ_u je hranice s Dirichletovskou podmínkou. Povrchové síly $P = (P_1, P_2)$ působí na hranici Γ_p, $P \in L^2(\Gamma_p)$. Těleso je zdola "podepřeno" podél hranice Γ_c (její tvar je určen řídící proměnnou $\boldsymbol{\alpha} \in \mathbb{R}^d$) tuhou překážkou. Množinu přípustných návrhových proměnných nazveme $\mathcal{U}_{ad}$. Na této hranici je předepsáno Trescovo tření s koeficientem tření závislým na řešení $\mathcal{F} : R_+ \to R_+$. Zavedeme si následující množinu

$$\mathcal{K}(\boldsymbol{\alpha}) := \{\boldsymbol{v} \in \mathbb{R}^n \,|\, \boldsymbol{v}_\nu \geq -\boldsymbol{\alpha}\}, \quad \boldsymbol{\alpha} \in \mathcal{U}_{ad}, \tag{1}$$

kde $\boldsymbol{v}_\nu$ odpovídá normálovému posunutí. Algebraická formulace diskrétního Signoriniho problému s Trescovým třením s koeficientem třením závislým na řešením je následující

$$\left.\begin{aligned}&\text{Najděte } \boldsymbol{u} \in \mathcal{K}(\boldsymbol{\alpha}) \text{ takové, že pro každé } \boldsymbol{v} \in \mathcal{K}(\boldsymbol{\alpha}) : \\ &\langle \mathbb{A}(\boldsymbol{\alpha})\boldsymbol{u}, \boldsymbol{v}-\boldsymbol{u}\rangle_n + \sum_{i=1}^{p} \omega_i(\boldsymbol{\alpha})\mathcal{F}(|(\boldsymbol{u}_\tau)_i|)\big(|(\boldsymbol{v}_\tau)_i| - |(\boldsymbol{u}_\tau)_i|\big) \geq \langle \boldsymbol{L}(\boldsymbol{\alpha}), \boldsymbol{v}-\boldsymbol{u}\rangle_n,\end{aligned}\right\} \tag{2}$$

kde $\mathbb{A} \in \mathbb{R}^{n\times n}$ a $\boldsymbol{L} \in \mathbb{R}^n$ jsou matice tuhosti a vektor sil závislé na řídící proměnné $\boldsymbol{\alpha}$.

Nyní si zavedeme vektor Lagrangeových multiplikátorů $\boldsymbol{\lambda} \in \mathbb{R}^p_+$ (p je počet kontaktních uzlů) pro omezení $\boldsymbol{v} \in \mathcal{K}(\boldsymbol{\alpha})$ a vektor $(\boldsymbol{u}, \boldsymbol{\lambda})$ nazveme stavovou proměnnou. Nyní zavedeme rozdělení vektoru posunutí $\boldsymbol{u}$ na $(\boldsymbol{u}_t, \boldsymbol{u}_\nu)$, kde $\boldsymbol{u}_t$ přísluší tečnému posunutí a $\boldsymbol{u}_\nu$ odpovídá normálovému posunutí. Dále zredukujeme naši úlohu a budeme se zabývat pouze kontaktními uzly. Stavová úloha realizuje zobrazení $\mathcal{S} : \boldsymbol{\alpha} \in \mathbb{R}^d \to (\boldsymbol{u}_t, \boldsymbol{u}_\nu, \boldsymbol{\lambda}) \in \mathbb{R}^{3p}$ (řídícímu vektoru $\boldsymbol{\alpha} \in U_{ad}$ je přiřazeno řešení kontaktní úlohy $(\boldsymbol{u}_t, \boldsymbol{u}_\nu, \boldsymbol{\lambda})$). Diskretizovanou stavovou úlohu lze ekvivalentně popsat zobecněnou rovností (podrobně v [1] a [2]).

$$\left.\begin{aligned}\mathbf{0} &\in \mathbb{A}_{\tau\tau}(\boldsymbol{\alpha})\boldsymbol{u}_\tau + \mathbb{A}_{\tau\nu}(\boldsymbol{\alpha})\boldsymbol{u}_\nu - \boldsymbol{L}_\tau(\boldsymbol{\alpha}) + Q_1(\boldsymbol{\alpha}, \boldsymbol{u}_\tau) \\ \mathbf{0} &= \mathbb{A}_{\nu\tau}(\boldsymbol{\alpha})\boldsymbol{u}_\tau + \mathbb{A}_{\nu\nu}(\boldsymbol{\alpha})\boldsymbol{u}_\nu - \boldsymbol{L}_\nu(\boldsymbol{\alpha}) - \boldsymbol{\lambda} \\ \mathbf{0} &\in \boldsymbol{u}_\nu + \boldsymbol{\alpha} + N_{\mathbb{R}^p_+}(\boldsymbol{\lambda}),\end{aligned}\right\} \tag{3}$$

kde multifunkce $Q_1 : \mathcal{U}_{ad} \times \mathbb{R}^p \rightrightarrows \mathbb{R}^p$ je definována jako:

$$\big(Q_1(\boldsymbol{\alpha}, \boldsymbol{u}_\tau)\big)_i := \omega_i(\boldsymbol{\alpha})\mathcal{F}(|(\boldsymbol{u}_\tau)_i|)\partial|(\boldsymbol{u}_\tau)_i| \quad \forall i = 1, \ldots, p, \tag{4}$$

a $N_{\mathbb{R}^p_+}(\cdot)$ je standardní normálový kužel.

3 Tvarová optimalizace pro kontaktní řlohu se zadaným třením s koeficientem tření závislým na řešení

Naším úkolem je nalézt řídící proměnnou $\boldsymbol{\alpha}$ určující Beziérovu funkci, kterou je modelována kontaktní hranice Γ_c, pro kterou nabývá cenový funkcionál $J(\boldsymbol{\alpha}, \mathcal{S}(\boldsymbol{\alpha}))$ svého minima. Úlohu diskrétní tvarové optimalizace zavedeme jako řešení

$$\min_{\boldsymbol{\alpha}\in U_{ad}} \mathcal{J}(\boldsymbol{\alpha}) = J(\boldsymbol{\alpha}, \mathcal{S}(\boldsymbol{\alpha})), \tag{5}$$

kde funkcionál J je spojitě diferencovatelný. K řešení této nehladké úlohy byla použita bundle trust metoda, která vznikla kombinací svazkových metod a trust region metody (podrobně viz [7]). Tato iterační metoda potřebuje rutinu, která v každém kroce vypočte hodnotu cenového funkcionálu (k tomu potřebujeme vyřešit stavovou úlohu) a jeden (libovolný) Clarkeův subgradient z Clarkeova zobecněného gradientu $\partial\mathcal{J}(\boldsymbol{\alpha})$. Pro jeho nalezení použijeme tvrzení

$$\partial\mathcal{J}(\boldsymbol{\alpha}) = \nabla_1 J(\boldsymbol{\alpha}, \mathcal{S}(\boldsymbol{\alpha})) + \operatorname{conv}\{\boldsymbol{C}^T\nabla_2 J(\boldsymbol{\alpha}, \mathcal{S}(\boldsymbol{\alpha}))|\boldsymbol{C} \in \partial\mathcal{S}(\boldsymbol{\alpha})\} \tag{6}$$

(viz [4]). Protože platí $\{\boldsymbol{C}^T\boldsymbol{y}^*|\boldsymbol{C} \in \partial\mathcal{S}(\boldsymbol{\alpha})\} = \operatorname{conv} D^*\mathcal{S}(\boldsymbol{\alpha})(\boldsymbol{y}^*)$ pro všechna $\boldsymbol{y}^*$, stačí nalézt jeden prvek z množiny $D^*\mathcal{S}(\boldsymbol{\alpha})(\nabla_2 J(\boldsymbol{\alpha}, \mathcal{S}(\boldsymbol{\alpha})))$. Prvky limitní koderivace $D^*\mathcal{S}(\boldsymbol{\alpha})$ najdeme použitím nehladkého kalkulu B. Morduchoviče (viz [6]). Podrobně v [3].

4 Numerický příklad

Pro numerické řešení stavové úlohy byla použita metoda postupných aproximací, kde každá iterace představuje Signoriniho úlohu se zadaným třením a daným koeficientem tření vypočteným z předchozí iterace. Numerické řešení stavové úlohy bylo implementováno (stejně jako celé řešení tvarově-optimalizační úlohy) v knihovně MatSol (viz [5]). Tato knihovna byla vyvinuta v prostředí Matlab.

Nyní použijeme navržený postup pro řešení následující úlohy:

$$\begin{array}{ll} \min & \|\bar{\boldsymbol{\lambda}} - \boldsymbol{\lambda}\|_2^2 \\ \text{s omezením} & \boldsymbol{\alpha} \in U_{ad}, \end{array} \tag{7}$$

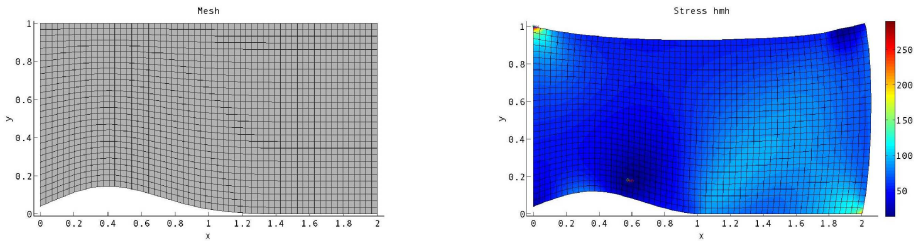
Předpokládejme, že koeficient tření $\mathcal{F}$ je popsán takto

$$\mathcal{F}(t) = 0.25 \cdot \frac{1}{t^2 + 1} \quad \forall t \in \mathbb{R}_+, \tag{8}$$

a mez skluzu je dána $g = 150$.

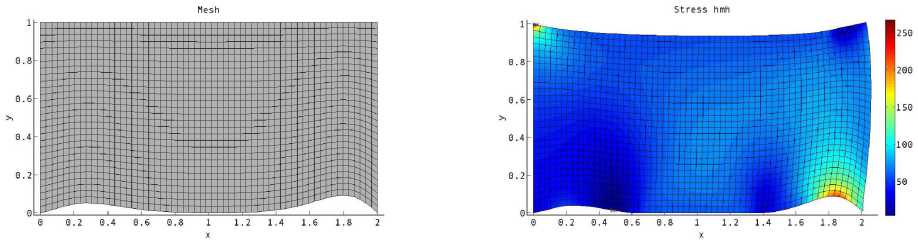
Naši oblast jsme nyní diskretizovali sítí s 1800 uzly, její velikost je 2x1. Povrchové tlaky na hranici Γ_p jsou předepsány takto $\boldsymbol{P}^1 = (0; -60 \text{ MPa})$ na $(0, 1.8) \times \{1\}$ a $\boldsymbol{P}^1 = (0; 0)$ na $(1.8, 2) \times \{1\}$, zatímco $\boldsymbol{P}^2 = (50 \text{ MPa}; 30 \text{ MPa})$ na $\{2\} \times (0, 1)$. Fyzikální parametry oblasti mají tyto hodnoty – Youngův modul $E = 1$ GPa a Poissonova konstanta $\nu = 0.3$. Dimenze návrhové proměnné $\boldsymbol{\alpha}$ řídící Beziérovu funkci, kterou je dána hranice Γ_c, je 20.

Počáteční návrh, jeho deformace a rozložení von Misesova redukovaného napětí je na obrázku 2.

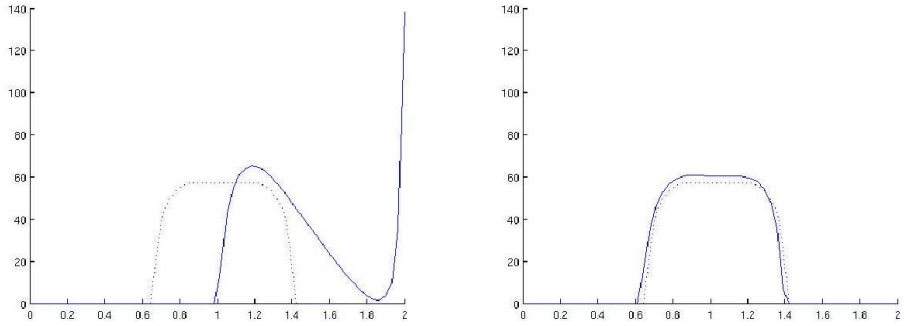


Obrázek 2: Počáteční návrh.

Obrázek 3 ukazuje optimalizovaný návrh, jeho deformaci a rozložení von Misesova redukovaného napětí.



Obrázek 3: Optimalizovaný návrh.



Obrázek 4: Rozložení normálového napětí na kontaktní hranici pro počáteční návrh (vlevo) a optimalizovaný návrh (vpravo).

Rozložení normálového napětí na kontaktní hranici (plná čára) i předepsané normálové napětí $\bar{\boldsymbol{\lambda}}$ (tečkovaná čára) pro počáteční i optimalizovaný tvar tělesa jsou zobrazeny na obrázku 4.

Hodnota cenového funkcionálu pro počáteční návrh je $5.910 \cdot 10^4$, zatímco hodnota cenového funkcionálu pro výsledný návrh je $9.1457 \cdot 10^2$.

Acknowledgement: Tato práce byla podpořena Evropským fondem regionálního rozvoje (ERDF) v rámci projektu Centra excelence IT4Innovations (CZ.1.05/1.1.00/02.0070) a projektem SPOMECH - Vytvoření multidisciplinárního vědeckovýzkumného týmu pro spolehlivé řešení úloh mechaniky, reg. č. CZ.1.07/2.3.00/20.0070 v rámci Operačního programu Vzdělávání pro konkurenceschopnost a financovaného ze strukturálních fondů EU a státního rozpočtu ČR.

References

[1] P. Beremlijski, J. Haslinger, M. Kočvara, J. Outrata: *Shape optimization in contact problems with Coulomb friction*. In: SIAM Journal on Optimization 12 (3), 2002, pp. 561–587.

[2] P. Beremlijski, J. Haslinger, M. Kočvara, R. Kučera, J. Outrata: *Shape optimization in three-dimensional contact problems with Coulomb friction*. In: SIAM Journal on Optimization 20 (1), 2009, pp. 416–444.

[3] P. Beremlijski, J. Haslinger, J. Outrata, R. Pathó: *Numerical solution of 2D contact shape optimization problems involving a solution-dependent coefficient of friction*. In: Springer (submitted).

[4] F. H. Clarke: *Optimization and nonsmooth analysis*. J. Wiley & Sons, 1983.

[5] T. Kozubek, A. Markopoulos, T. Brzobohatý, R. Kučera, V. Vondrák, Z. Dostál: *MatSol – MATLAB efficient solvers for problems in engineering*. `http://www.am.vsb.cz/matsol`.

[6] B. S. Mordukhovich: *Variational analysis and generalized differentiation*, Volumes I and II. Springer-Verlag, 2006.

[7] J. Outrata, M. Kočvara, J. Zowe: *Nonsmooth approach to optimization problems with equilibrium constraints: theory, applications and numerical results*. Kluwer Acad. Publ., 1998.

Spiders on the vineyard

M. Biák, D. Janovská

Department of Mathematics, Institute of Chemical Technology, Prague

1 Introduction

The population model of the spiders hunting the insect on the vineyard can be described as the set of ordinary differential equations, see [1]. This type of model is known as the predator–prey model. We show how to integrate a human intervention into this model. We formulate Filippov system that includes both cases - with and without the intervention. Then we analyze this model using the theory described in [2].

All simulations are performed in modified version of the program developed by Petri T. Piiroinen and Yuri A. Kuznetsov, see [3] and [4].

2 Model equations

The predator–prey model of spiders and insect on the vineyard is described as the set of ordinary differential equations:

$$\dot{f} = rf(1-\frac{f}{W}) - csf, \tag{1}$$

$$\dot{s} = s(-a+\frac{kbv}{H+v}+kcf), \tag{2}$$

$$\dot{v} = v(e-\frac{bs}{H+v}), \tag{3}$$

where $v(t)$ is the population of the insect on the vineyard, $f(t)$ is the population of the insect outside the vineyard, $s(t)$ is the population of the spiders. If man intervenes into the ecosystem by spraying to prevent an overgrowth of insects, equations (1)–(3) pass to

$$\dot{f} = rf(1-\frac{f}{W}) - csf - h(1-q)f, \tag{4}$$

$$\dot{s} = s(-a+\frac{kbv}{H+v}+kcf) - hKqs, \tag{5}$$

$$\dot{v} = v(e-\frac{bs}{H+v}) - hqv, \tag{6}$$

where an extra term in each equation represents the mortality caused by spraying. All parameters in (1)–(3) and (4)–(6) are positive real numbers.

The question is how to introduce the model, that includes both cases (with and without spraying) and that keeps the population of the insect on the vineyard below a given limit. We will show that such model is a type of Filippov system and it can be treated using the techniques stated e.g. in [2].

3 Population model as Filippov system

Because only the positive values of $f(t)$, $s(t)$, $v(t)$ have a physical meaning, we start with a region $\mathcal{D} = \{(f, s, v) \in \mathbb{R}^3 : f(t) > 0, s(t) > 0, v(t) > 0\}$. Let us have a scalar function $\varphi : \mathcal{D} \to \mathbb{R}$. The function φ divides the region $\mathcal{D}$ into:

$$\begin{aligned} S_1 &= \{\mathbf{x} \in \mathcal{D} : \varphi(\mathbf{x}) > 0\}, \\ S_2 &= \{\mathbf{x} \in \mathcal{D} : \varphi(\mathbf{x}) < 0\}, \\ \Sigma &= \{\mathbf{x} \in \mathcal{D} : \varphi(\mathbf{x}) = 0\}, \end{aligned}$$

where $\mathbf{x} = (f, s, v)^T$.

In our case, we want to keep the population $v(t)$ of the insect on the vineyard below the given value $v_m \in \mathbb{R}$, $v_m > 0$. Therefore, our function $\varphi(\mathbf{x})$ will be

$$\varphi(f, s, v) = v_m - v. \tag{7}$$

We define a Filippov system on $\mathcal{D} = S_1 \cup S_2 \cup \Sigma$

$$\mathcal{F} : \ \dot{\mathbf{x}} = \begin{cases} \mathbf{g}^{(1)}(\mathbf{x}) \ , & \mathbf{x} \in S_1, \\ \mathbf{g}^{(0)}(\mathbf{x}) \ , & \mathbf{x} \in \Sigma, \\ \mathbf{g}^{(2)}(\mathbf{x}) \ , & \mathbf{x} \in S_2, \end{cases} \tag{8}$$

where $\dot{\mathbf{x}} = (\dot{f}, \dot{s}, \dot{v})^T$, and where the vector fields $\mathbf{g}^{(i)} : \mathbb{R}^3 \to \mathbb{R}^3$, $i = 1, 2$, are

$$\mathbf{g}^{(1)} = \begin{pmatrix} rf(1 - \frac{f}{W}) - csf \\ s(-a + \frac{kbv}{H+v} + kcf) \\ v(e - \frac{bs}{H+v}) \end{pmatrix}, \quad \mathbf{g}^{(2)} = \begin{pmatrix} rf(1 - \frac{f}{W}) - csf - h(1-q)f \\ s(-a + \frac{kbv}{H+v} + kcf) - hKqs \\ v(e - \frac{bs}{H+v}) - hqv \end{pmatrix}.$$

If $\varphi(f, s, v) > 0$, no spraying occurs and the vector field $\mathbf{g}^{(1)}(\mathbf{x})$ is in effect. If the population $v(t)$ of the insect on the vineyard rises above a given value v_m, i.e. if $\varphi(f, s, v) < 0$, the spraying begins and the vector field $\mathbf{g}^{(2)}(\mathbf{x})$ takes place. The spraying goes on, until the value of $v(t)$ decreases below v_m, when $\mathbf{g}^{(1)}(\mathbf{x})$ applies again.

Before we define the vector field $\mathbf{g}^{(0)}(\mathbf{x})$ that determines behavior of the system (8) on the boundary Σ, we need to distinguish two types of sets on Σ. We define a scalar function $\sigma : \Sigma \to \mathbb{R}$,

$$\sigma(\mathbf{x}) = \langle \nabla\varphi, \mathbf{g}^{(1)} \rangle \langle \nabla\varphi, \mathbf{g}^{(2)} \rangle,$$

and we obtain two sets on Σ, the crossing set $\Sigma_c \subseteq \Sigma = \{\mathbf{x} \in \Sigma : \varphi(\mathbf{x}) = 0 \ \wedge \ \sigma(\mathbf{x}) > 0\}$, and the sliding set $\Sigma_s \subseteq \Sigma = \{\mathbf{x} \in \Sigma : \varphi(\mathbf{x}) = 0 \ \wedge \ \sigma(\mathbf{x}) \leq 0\}$,

In our case, the scalar function $\sigma(f, s, v)$ reads

$$\begin{aligned} \sigma(f, s, v) &= \langle \nabla\varphi, \mathbf{g}^{(1)} \rangle \langle \nabla\varphi, \mathbf{g}^{(2)} \rangle, \\ \langle \nabla\varphi, \mathbf{g}^{(1)} \rangle &= \left(\frac{bs}{H + v_m} - e \right), \\ \langle \nabla\varphi, \mathbf{g}^{(2)} \rangle &= \left(\frac{bs}{H + v_m} - e + hq \right). \end{aligned}$$

If $\sigma(f,s,v) \leq 0$, trajectory slides along the sliding set Σ_s. If $\sigma(f,s,v) > 0$, we are on the crossing set Σ_c and trajectory leaves the boundary.

On Σ_c, we put

$$\mathbf{g}^{(0)} = \frac{1}{2}\left(\mathbf{g}^{(1)} + \mathbf{g}^{(2)}\right).$$

For $\mathbf{x} \in \Sigma_s$, we define a smooth vector field $\mathbf{g}^{(0)} : \mathbb{R}^3 \to \mathbb{R}^3$,

$$\mathbf{g}^{(0)} = \lambda \mathbf{g}^{(1)} + (1-\lambda)\mathbf{g}^{(2)}, \quad \lambda = \frac{\langle \nabla\varphi, \mathbf{g}^2 \rangle}{\langle \nabla\varphi, \mathbf{g}^2 - \mathbf{g}^1 \rangle}, \tag{9}$$

where $\lambda \in \mathbb{R}$, $0 \leq \lambda \leq 1$.

The points in which $\sigma(f,s,v) = 0$ are called tangent points. There are two sets of tangent points T_1 and T_2 on the boundary Σ:

$$\begin{aligned} \mathrm{T}_1 &= \{(f,s,v) : f > 0, s = \frac{1}{b}e(H+v_m), v = v_m\}, \\ \mathrm{T}_2 &= \{(f,s,v) : f > 0, s = \frac{1}{b}(e-hq)(H+v_m), v = v_m\}. \end{aligned}$$

Let us assume that $e < hq$, i.e. $s_{T_2} < 0$. The sets T_1 and T_2 delimit the sliding set Σ_s, and due to the fact that $s_{T_2} < 0 < s_{T_1}$, the sliding set $\Sigma_s \subset \Sigma = \{(f,s,v) : f > 0 \wedge 0 < s \leq s_{T_1} \wedge v = v_m\}$ is a semi-infinite stripe with the non-zero width equal to $\frac{H+v_m}{b}hq$.

If in (9) $\mathbf{g}^{(0)}(P) = 0$, the point P is a pseudo-equilibrium of the Filippov system.

We performed simulations with the parameters listed in Table 1. We found that a local sliding bifurcation occurs for the value $v_m = 0.9$. During the simulations we observed a global bifurcation, too.

Parameter	Value	Meaning
a	0.2	specific mortality rate of predators
b	1.18	specific reproduction rate of predators per 1 prey eaten in the vineyards
c	0.2	specific reproduction rate of predators per 1 prey eaten in the woods
e	0.5	specific growth rate of the prey in the vineyards
r	1	specific growth rate of the prey in the woods
k	1	conversion factor of prey into new spiders, $k \leq 1$
H	7	carrying capacity of the vineyard
W	1	carrying capacity of the woods
h	0.6	effectiveness of the insecticide against the parasites
K	0.01	smaller effect the insecticide should have on the spiders, $0 < K < 1$
q	0.9	portion of insecticide sprayed directly on the vineyards
$1-q$	0.1	portion which may accidentally be dispersed in the woods
v_m	0.3–3.0	limit of the population of the insect on the vineyard

Table 1: The parameters used for the simulation of the system $\mathcal{F}$.

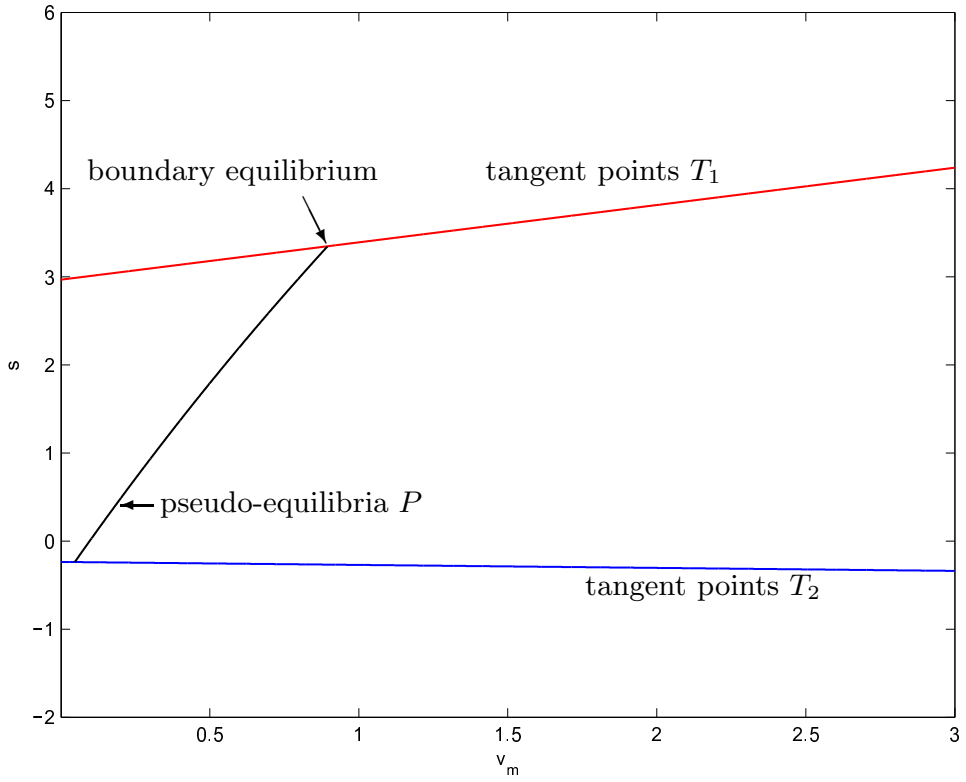


Figure 1: Solution diagram.

4 Conclusions

In the population model of the spiders on the vineyard, we discovered both local and global sliding bifurcation. The local bifurcation is caused by a collision of the equilibrium with the boundary Σ and is called boundary–equilibrium bifurcation. The global sliding bifurcation in the simulations is caused by a collision of the periodic trajectory with the boundary Σ.

Acknowledgement: The work is a part of the research project MSM 6046137306 financed by MSMT, Ministry of Education, Youth and Sports, Czech Republic.

References

[1] E. Venturino, M. Isaia, F. Bona, S. Chatterjee, G. Badino: *Biological controls of intensive agroecosystems: Wanderer spiders in the Langa Astigiana.* Ecological Complexity 5 (2), 2008, pp. 157–164.

[2] M. di Bernardo, C. J. Budd, A. R. Champneys, P. Kowalczyk: *Piecewise-smooth dynamical systems: theory and applications.* Springer-Verlag, London, 2008.

[3] P. T. Piiroinen, Yu. A. Kuznetsov: *An event-driven method to simulate Filippov systems with accurate computing of sliding motions.* ACM Trans. Math. Software 34 (13), 2008, pp. 1–24.

[4] F. Dercole, Yu. A. Kuznetsov: *SlideCont: An AUTO97 driver for sliding bifurcation analysis.* ACM Trans. Math. Software 31, 2005, pp. 95–119.

Iterative solution of singular systems with applications

R. Blaheta, O. Jakl, J. Starý, E. Turan

[1,2,3] Institute of Geonics AS CR, Ostrava
[4] ETH, Zurich

Dedicated to Professor Ivo Marek on the occasion of his 80th birthday.

1 Introduction

This contribution concerns the iterative solution of singular systems which arise in many applications. Let us mention the following

- solution of PDE problems with pure Neumann boundary conditions (which is our main aim), see [7], [8], [20]. Such problems have a specific role in numerical upscaling, see [6],
- solution of Neumann type subproblems in domain decomposition techniques as FETI, Neumann-Neumann, BDDC methods, see [22], [16],
- analysis of Markov chain problems, computation of stochastic vector, see e.g. [18], [19],
- computer tomography [15], [14] and inverse problems [4], [21].

2 Iterative solution of singular symmetric semidefinite systems

Let us focus on iterative solution of linear systems of the form

$$Au = b, \tag{1}$$

where A is a singular, symmetric, positive semidefinite $n \times n$ matrix, $b \in R^n$. For $u, v \in R^n$ denote $\langle u, v\rangle = u^T v$ and $\|u\|$ the Eucledian inner product and norm. Due to symmetry of A, the range $R(A)$ and the null space $N(A)$ are mutually orthogonal with respect to the Eucledian inner product and the vectors $u \in R^n$ can be uniquely decomposed as

$$u = u_N + u_R, \text{ where } u_N \in N(A) \text{ and } u_R \in R(A).$$

Let $b = b_N + b_R$, then the system (1) has infinitely many generalized (least squares) solutions u,

$$\|Au - b\| = \min\{\|Av - b\|,\ v \in R^n\} \tag{2}$$

among which there is a unique least squares solution u^* with the minimal Eucledian norm. Note that $u^* = A^+ b$, where A^+ is the Moore-Penrose pseudoinverse of A, see [10], [16]. If $b \in R(A)$, i.e. the system (1) is consistent, then the generalized solutions are standard solutions of (1).

Let us assume that (1) is solved iteratively with denoting the i-th iteration u^i,

$$u^i \in u^0 + K_i(A, r^0) = u^0 + \text{span}\{r^0, Ar^0, \ldots, A^{i-1}r^0\},\ r^0 = b - Au^0, \tag{3}$$

where $K_i(A, r^0) = \text{span}\{r^0, Ar^0, \ldots, A^{i-1}r^0\}$ is a Krylov space. Then

$$u^i = u^0 + q_{i-1}(A)r^0, \text{ where } q_{i-1} \text{ is a polynomial of order } \leq i - 1. \tag{4}$$

The convergence can be investigated through behaviour of $e^i = u^i - u^*$. If $e^i \to 0$ then the iterations converge to the minimal least squares solution u^*. If $e^i \to w$, where $w \in N(A)$, then the iterations converge to a (generalized) solution of A.

From (4), it follows that

$$e^i = u^0 - u^* + q_{i-1}(A)(b_N + Au^* - Au^0) = p_i(A)e^0 + q_{i-1}(0)b_N, \tag{5}$$

where $p_i(\lambda) = 1 - \lambda q_{i-1}(\lambda)$.

If $e^0 = e^0_N + e^0_R$ then $p_i(A)e^0_N = u^0_N$ and $p_i(A)e^0_R$ depends on values $p_i(\lambda)$ on $\lambda \in \sigma(A) \setminus \{0\}$. The second term is zero for consistent problems, but otherwise can be convergent if $q_{i-1}(0) = -p_i'(0) \neq 0$.

The simplest Richardson's iteration $u^{i+1} = u^i + \omega A(b - u^i)$ fulfill (3), (4), (5) with

$$p_i(\lambda) = (1 - \omega\lambda)^i, \quad p_i(0) = 1, \quad q_{i-1}(0) = -p_i'(0) = (i+1)\omega.$$

Thus, the method converges ($e^0 \to u^0_N$) for the consistent problems, but diverges (the second terms gradually dominates) for the inconsistent case ($b_N \neq 0$).

To get convergence even for inconsistent case, the method needs a modification. For example, we can use extrapolation of Richardson's iterations [17]. For

$$\bar{u}^{i+1} = u^{i+1} - (i+1)(u^{i+1} - u^i),$$

we get

$$\begin{aligned} \bar{u}^{i+1} - u^* &= u^{i+1} - u^* - (i+1)(u^{i+1} - u^i) = p_{i+1}(A)e^0 + (i+1)\omega(b_N + A(u^* - u^i)) \\ &= p_{i+1}(A)e^0 + (i+1)\omega Ae^i = p_{i+1}(A)e^0 + (i+1)\omega A(p_i(A)e^0 + i\omega b_N)) \\ &= p_{i+1}(A)e^0 + (i+1)\omega(p_i(A)Ae^0)). \end{aligned}$$

This extrapolated method converges since $p_i(\lambda) \leq q^i$ for all $\lambda \in \sigma(A) \setminus \{0\}$, where $q < 1$ for a suitable ω.

It means that there are ways how to damp the divergence of the null space component of the iterations. On the other hand, this divergence in the null space component may not cause a problem in case that we are interested only in quantities, which do not depend on the null space component, like gradients, fluxes, strains and stresses.

A similar analysis can be done for other iterative methods applied to singular systems, see e.g. [10]. For the conjugate gradient (CG) method, the convergence can be proven in the consistent case, see eg. [1]. But the inconsistence influence both $N(A)$ and $R(A)$ components of the iterations, see [13], [7] and the next section.

3 Solution of Neumann problems

The solution of boundary value problems with pure Neumann boundary conditions arises in different applications, see the other sections. If the solution of the continuous Neumann problem exists, then global balance (consistency) conditions like (7) are satisfied. On the contrary, these conditions guarantee the existence of the (not unique) solution. For example ([20], [7]), for the Neumann problem,

$$-\mathrm{div}(\nabla u) = f \text{ in } \Omega \quad \text{and} \quad \nabla u \cdot n = g \text{ in } \partial\Omega \tag{6}$$

the solution exits if and only if

$$\int_\Omega f\,dx + \int_{\partial\Omega} g\,dx = 0. \tag{7}$$

In the case (6), (7), if u is a solution, then $u+v$ is a solution for all $v \in \mathcal{N} = \text{span}\{1\}$, where 1 is a constant function in Ω. A finite element discretization then should provide a consistent singular linear system (1) with the nullspace $N(A) = \mathcal{N}_h$ provided by discretization of $\mathcal{N}$. However, the computer arithmetic and numerical integration errors may cause that the FEM system is inconsistent and/or $N(A) \neq \mathcal{N}_h$.

Problems with inconsistency and singularity can be treated by using a priori knowledge about $\mathcal{N}$ and $\mathcal{N}_h$. For example, we are able to regularize the problem by fixing some degrees of freedom and solving the problem $R_{dof}AR_{dof}^T u = R_{dof}b$ instead of (1). Here, R_{dof} is the restriction opperator omitting the fixed DOF's. Such a technique is frequently used in engineering community, but without a special care [9] the modified system matrix $R_{dof}AR_{dof}^T$ can be very ill-conditioned which is a serious drawback for the iterative solution.

Using the knowledge of $\mathcal{N}$, other techniques use the projection $P : R^n \to \mathcal{R}_h$, where $\mathcal{R}_h$ is the orthogonal complement of $\mathcal{N}_h$. The projector can be constructed as $P = I - V(V^TV)^{-1}V^T$, where V is a matrix, whose columns create a basis of $\mathcal{N}_h$. Such projector can be applied within any iterative method. In *PCGstab1* algorithm, the projection P is used to project the right hand side vectors or all residuals during the PCG iterative process. In *PCGstab2*, the projection P is applied twice per iteration to project both residuals and computed iterations. Figure shows these stabilizations of the PCG method. *PCGstab2* is equivalent to the replacement of A by PAP which also makes the system matrix singular. The fully stabilized *PCGstab2* was introduced e.g. in [11]. Note that $g = G(r)$ denotes the action of preconditioner, which can be also nonlinear (variable, flexible).

given u^0
compute $r^0 = P_a(b - Au^0)$, $g^0 = P_bG(r^0)$, $v^0 = g^0$
for $i = 0, 1, \ldots$ **until** convergence **do**
  $w^i = P_cAP_dv^i$
  $\alpha_i = \langle r^i, g^i\rangle / \langle w^i, v^i\rangle$
  $u^{i+1} = u^i + \alpha_i v^i$
  $r^{i+1} = P_a(r^i - \alpha_i w^i)$
  $g^{i+1} = P_bG(r^{i+1})$
  $\beta_{i+1} = \langle g^{i+1}, r^{i+1}\rangle / \langle g^i,\, r^i\rangle$
  $v^{i+1} = g^{i+1} + \beta_{i+1}v^i$
end

a) Standard PCG:
$P_a = P_b = P_c = P_d = I$

b) *PCGstab1*:
$P_a = P$
$P_b = P_c = P_d = I$

c) *PCGstab2*:
$P_a = P_b = P$
$P_d = P_c = I$
or equivalently
$P_a = P_b = I$
$P_c = P_d = P$

Figure: PCG algorithms.

Note that an application of PCG to inconsistent system is problematic from two reasons. The inconsistent part of the right hand side enters the $N(A)$-part of the iterations and can make them divergent, but the inconsistent part also enters the formulas for α and β and spoils the $R(A)$-part of the iterations, see [13], [5].

4 Application in upscaling

The elastic response of a representative volume Ω is characterized by homogenized elasticity C or compliance S tensors ($S = C^{-1}$). The compliance tensor can be determined from the relation

$$S\langle\sigma\rangle = S\sigma_0 = \langle\varepsilon\rangle, \tag{8}$$

where $\langle\sigma\rangle$ and $\langle\varepsilon\rangle$ are volume averaged stresses and strains computed from the Neumann problem

$$-\mathrm{div}(\sigma) = 0, \quad \sigma = C_m\varepsilon, \quad \varepsilon = (\nabla u + (\nabla u)^T)/2 \quad \text{in } \Omega, \tag{9}$$

$$\sigma n = \sigma_0 n \quad \text{on } \partial\Omega. \tag{10}$$

Above, σ and ε denote stress and strain in the microstructure, C_m is the variable local elasticity tensor, u and n denote the displacement and the unit normal, respectively. The use of Neumann boundary conditions allows us to get a lower bound for the upscaled elasticity tensor [6].

In analysis of geocomposites (see [6]), the domain Ω is a cube with a relatively complicated microstructure. The FEM mesh is constructed on the basis of CT scans. Consequently using the GEM software [3], the domain is discretized by linear tetrahedral finite elements. The arising singular system is then solved by stabilized *PCGstab1* method implemented in different software and using various preconditioners:

GEM-DD is a solver fully implemented in GEM software. It uses one-level additive Schwarz domain decomposition preconditioner with subproblems replaced by displacement decomposition incomplete factorization described in [2]. The resulting preconditioner is symmetric positive definite.

GEM-DD-CG solver differs in preconditioning, which is a two-level Schwarz domain decomposition arising from the previous GEM-DD by additive involvement of a coarse problem correction. The coarse problem is created by a regular aggregation of $6 \times 6 \times 3$ nodes with 3 DOF's per aggregation. In this case, the coarse problem is singular with a smaller null space containing only the rigid shifts. The coarse problem is solved only approximately by inner (not stabilized) CG method with a lower solution accuracy - relative residual accuracy $\varepsilon_0 \leq 0.01$.

Trilinos ILU is solver running in Trilinos, where the system from GEM is imported. The preconditioner is similar to GEM-DD, i.e. one-level Schwarz with the minimal overlap and working on the same subdomains as in GEM-DD are used. The subproblems are replaced by ILU without displacement decomposition, using a drop tolerance and a fill limit.

Trilinos ML-DD is again running in TRILINOS and uses multilevel-level V-cycle preconditioner exploiting smoothed aggregations with aggressive coarsening, see [12]. Six DOF's translational plus rotational are used per aggregation. ILU is applied as smoother at the finest level, other smoothing is realised by symmetrized Gauss-Seidel. The coarsest problem is solved by a direct solver.

	GEM						**Trilinos**					
	DD			*DD+CG*			*ILU*			*ML-DD*		
# Sd	# It	T_{prep}	T_{iter}	# It	T_{prep}	T_{iter}	# It	T_{prep}	T_{iter}	# It	T_{prep}	T_{iter}
1							345	224.5	2672.4		×	
2	293	0.3	541.4	137	20.1	256.4	472	135.9	1628.3	43	813.6	804.5
4	302	0.2	302.2	124	20.0	125.9	463	112.5	1022.6	46	445.6	404.9
8	300	0.1	175.3	115	19.9	75.7	441	85.9	517.6	53	302.9	203.8
16	350	0.1	148.5	116	19.9	73.6	387	75.4	443.9	57	335.4	146.9

Table: Solution of the Neumann problem in elasticity, slightly more than 6 million mil DOF's, stopping criterion $\|r\|/\|rhs\| \leq \varepsilon = 10^{-5}$. Numbers of iterations (#It), wall-clock time in seconds for solver preparation (T_{prep}) and time for performing the iterations (T_{iter}) are provided for various numbers of subdomains (# Sd; always corresponding to the number of employed processing units). GEM solvers have not the single processor mode, the ML-DD solver ended on single processor with the message "Not enough space for domain decomposition"(×).

The parallel computing was performed on 32 core NUMA machine at the Institute of Geonics with eight quad-core AMD Opteron 8830/2.5 GHz processors and 128 GB of DDR2 RAM. Because of using stabilized PCG and also because we were interested only on strains and stresses, we concentrate on $R(A)$-part of the solution and watch in Table only the convergence in the residual norm.

We can see that the stabilized CG works well. On the other hand the unstabilized version converge up to a smaller residual tolerance $\varepsilon = 0.01 - 0.001$ and then started to blow up, see [5]. It indicates that numerical consistency and numerical singularity are not enough, which was a bit unexpected in our case as we used lowest order linear finite elements and problem with piecewise constant boundary condition, so that the adopted numerical integration should be exact. On the other hand, the systems were assembled in single precision.

5 Conclusions

The aim of this contribution was to show techniques for efficient solution of singular symmetric positive semidefinite problems. We can see that the stabilized PCG is a good choice for systems arising from the numerical solution of Neumann problems, or more generally problems with a known small dimensional null space. There are also other possibilities of stabilization as e.g. the use of additive regularization.

The second aim was a comparison of specialized solvers from the in-house finite element software GEM and more general solvers from the Trilinos library. We provided some comparison while this work is still continuing.

Acknowledgement:

This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070).

References

[1] O. Axelsson: *Iterative solution methods*. Cambridge University Press, 1994.

[2] R. Blaheta: *Displacement decomposition – incomplete factorization preconditioning techniques for linear elasticity problems*. Numerical Linear Algebra with Applications 1, 1994, pp. 107–128.

[3] R. Blaheta, O. Jakl, R. Kohut, J. Starý: *GEM – A Platform for Advanced Mathematical Geosimulations*. In: R. Wyrzykowski et al. (eds.): PPAM 2009, Part I, LNCS 6067, 2010, pp. 266–275.

[4] R. Blaheta, P. Byczanski, M. Čermák, R. Hrtus, R. Kohut, A. Kolcun, J. Malík, S. Sysala: *Analysis of Äspö Pillar Stability Experiment: Continuous TM Model Development and Calibration*. Journal of Rock Mechanics and Geotechnical Engineering, Special Issue Decovalex 2011, to appear.

[5] R. Blaheta, O. Jakl, J. Starý, E. Turan: *Parallel solvers for numerical upscaling*. Proceedings of the Workshop on the State-of-the-Art in Scientific and Parallel Computing PARA 2012, LNCS, Springer, accepted.

[6] R. Blaheta, R. Kohut, A. Kolcun, K. Souček, L. Staš, L. Vavro: *Digital image based numerical micromechanics of geocomposites with application to chemical grouting*, submitted to Int. J. Rock Mech. Min. Sci. 2012.

[7] P. Bochev, R.B. Lehoucq: *On the finite element solution of the pure Neumann problem.* SIAM Review 47, 2005, pp. 50-66.

[8] P. Bochev, R.B. Lehoucq: *On the finite element solution of the pure Neumann problem.* Comput. Meth. Appl. Math. 5, 2011, pp. 1–15.

[9] T. Brzobohatý, Z. Dostál, T. Kozubek, A. Markopoulos, P. Kovář: *Cholesky–SVD decomposition with fixing nodes to stable computation of a generalized inverse of the stiffness matrix of a floating structure.* Int. J. Numer. Meth. Engng. 88, 2011, pp. 493–509.

[10] M. Eiermann, I. Marek, W. Niethammer: *On the solution of singular linear systems of algebraic equations by semiiterative methods.* Numer. Math. 53, 1988, pp. 265–283.

[11] C. Farhat, J. Mandel, F.X. Roux: *Optimal convergence properties of the FETI domain decomposition method.* Computer Methods in Applied Mechanics and Engineering 115, 1994, pp. 367–388.

[12] M.W. Gee, Ch.M. Siefert, J.J. Hu, R.S. Tuminaro, M.G. Sala: *ML 5.0 Smoothed Aggregation User's Guide.* Tech. Report SAND2006-2649, Sandia National Lab., 2006.

[13] E.F. Kaasschieter: *Preconditioned conjugate gradients for solving singular systems.* Journal of Computational and Applied Mathematics 24, 1988, pp. 265–275.

[14] H. Köstler, C. Popa, S. Bergler, U. Rüde: *Algebraic multigrid for general inconsistent linear systems: The correction step.* Rep. 06-4, Lehrstuhl fur Informatik 10 (Systemsimulation), FAU Erlangen-Nurnberg, 2006.

[15] H. Köstler, C. Popa, M. Prümmer, U. Rüde: *Towards an algebraic multigrid method for tomographic image reconstruction.* In: P. Wesseling, E. Onate, J. Periaux (eds.): European Conference on Computational Fluid Dynamics ECCOMAS CFD 2006, TU Delft, The Netherlands, 2006.

[16] R. Kučera, T. Kozubek, A. Markopoulos, J. Machalová: *On the MoorePenrose inverse in solving saddle-point systems with singular diagonal blocks.* Numerical Linear Algebra with Applications 19, 2012, pp. 677–699.

[17] G.I. Marchuk: *Methods of numerical mathematics.* Springer-Verlag, New York, Heidelberg, Berlin, 1982. Czech transl. Academia, 1987.

[18] I. Marek, D.B. Szyld: *Iterative and semi-iterative methods for computing stationary probability vectors of Markov operators.* Math. Comp. 61, 1993, pp. 719–731.

[19] I. Marek, D.B. Szyld: *Algebraic Schwarz methods for the numerical solution of Markov chains.* Linear Algebra and its Applications 386, 2004, pp. 67–81.

[20] J. Nečas, I. Hlaváček: *Mathematical Theory of Elastic and Elastico-Plastic Bodies.* Studies in applied mechanics 3, Elsevier, 1981, 342 pages.

[21] A. Neumaier: *Solving Ill-Conditioned And Singular Linear Systems: A Tutorial On Regularization.* SIAM Review 40, 1998, pp. 636–666.

[22] A. Toselli, O. Widdlund: *Domain Decomposition Methods - Algorithms and Theory.* Springer-Verlag, Berlin, 2005.

Numerical schemes for lower urinary tract flow modelling

M. Brandner, J. Egermaier, H. Kopincová, J. Rosenberg

[1,2,3] NTIS – New Technologies for Information Society, University of West Bohemia, Pilsen
[4] New Technologies-Research Centre, University of West Bohemia, Pilsen

1 Introduction

The voiding is a very complex process. It consists of the transfer of information about the state of the bladder filling in to the spinal cord. Next part is the sending of the action potentials to the smooth muscle cells of the bladder. Even this process is not simple and includes the spreading of the action potential along the nerve axon and the transmission of the mediator (Ach – acetylcholine) in the synapse. The action potential starts the process of the smooth muscle contraction. The sliding between actin and myosin causing the change of the form (length) of the muscle cell and its stiffness can be observed as a kind of growth and remodeling. This approach described e.g. in [6] is used in this model. To be able to describe the very complex processes in the SMC in the efficient form it is necessary to use the irreversible thermodynamics. This approach was described in [7].

2 Bladder contraction

The whole model of the bladder contraction consists of the following parts:

- Model of the time evolution of the Ca^{2+} concentration. The Ca^{2+} intracellular concentration is the main control parameter for the next processes and finally for the smooth muscle contraction. Its increase depends on the flux $J_{agonist}$ of the mediator (in this case acetylcholine) via the nerve synapse.

$$\begin{aligned}
\frac{dc}{dt} &= J_{IP3} - J_{VOCC} + J_{Na/Ca} - J_{SRuptake} + J_{CICR} - J_{extrusion} + J_{leak} + J_{stretch} \\
\frac{ds}{dt} &= J_{SRuptake} - J_{CICR} - J_{leak} \\
\frac{dv}{dt} &= \gamma(-J_{Na/K} - J_{Cl} - 2J_{VOCC} - J_{Na/Ca} - J_K - J_{stretch}) \\
\frac{dw}{dt} &= \lambda K_{activate} \\
\frac{dI}{dt} &= J_{agonist} - J_{degrad},
\end{aligned} \tag{1}$$

  where the unknown functions represents: $c = c(t)$ calcium concentration in cytoplasm, $s = s(t)$ calcium concentration in ER/SR, $v = v(t)$ membrane tension, $w = w(t)$ probability of opening channels activated by Ca^{2+} and $I = I(t)$ IP3 sensitive reservoirs concentration in cytoplasm. For details and complete description of the functions and parameters see [4].

- Model of the time evolution of the phosphorylation of the light myosin chain. The muscle cell contraction is caused by the relative movement of the myosin and actin filaments. For

this it is necessary that the phosphorylation of the mentioned light myosin chain on the heads of the myosin occurs.

$$\begin{aligned} \frac{dA_M}{dt} &= k_5 A_{M_p} - (k_7 + k_6) A_M, \\ \frac{dA_{M_p}}{dt} &= k_3 M_p + k_6 A_M - (k_4 + k_5) A_{M_p}, \\ \frac{dM_p}{dt} &= k_1 (1 - A_M) + (k_4 - k_1) A_{M_p} - (k_1 + k_2 + k_3) M_p, \end{aligned} \tag{2}$$

where the unknown functions represent the following: $A_M = A_M(t)$ connected cross-bridges, $A_{M_p} = A_{M_p}(t)$ connected phosphorylated cross-bridges and $M_p = M_p(t)$ unconnected phosphorylated cross-bridges. $k_6 = k_6(c)$, the other terms k_i are constant. For details and complete description of the functions and parameters see [3]. Knowing this process also the time evolution of the ATP consumption (J_{cycl}) can be determined. The ATP (adenosintriphosphate) is the main energy source for the muscle contraction.

$$\frac{dY}{dt} = -Q_Q Y + L J_{cycl}, \tag{3}$$

where $Y = Y(t)$ represents the ATP concentration, Q_Q is the damping parameter and L is the constant.

- Model of the own contraction based on the GRT and the irreversible thermodynamics. The growth and remodelling theory [2] together with the laws of irreversible thermodynamics with internal variables was applied in [7] to describe the mechano-chemical coupling of the smooth muscle cell contraction. The product of the chemical reaction affinity (the ATP hydrolysis) with its rate plays an important role in the discussed model. Further it can be assumed that the rate of the ATP hydrolysis depends on the ATP consumption. The corresponding equations in the non-dimensional form are following:

$$\dot{x} = k_1 \left[\tau - z(x-1)\right], \quad \dot{y} = \frac{y}{k_2}\left[x\tau - \frac{1}{2}z(x-1)^2 + C'\right], \quad \dot{z} = \mathrm{sgn}(m)\cdot\left[r - \frac{1}{2}z(x-1)^2\right], \tag{4}$$

where $x = \frac{l}{l_r}$, $y = \frac{l_r}{l_0}$, l_0 is the initial length of the muscle fibre, l_r its length after stimulation when the fibre is unloaded (s. c. resting length), l the actual length (when the contraction is isometric this is the input value), τ the stress and k is the fibre stiffness, m and r are constants. The non-dimensional values are labeled with the single quote mark. The others symbols are the parameters.

3 Bladder and voiding model

To model the contraction of the bladder during the voiding process we will use the very simple model according [5]. The bladder is modelled as a hollow sphere with the output corresponding to the input into urethra. For the pressure in the bladder the following formula is introduced in [5]

$$p = \frac{V_{sh}}{3V}\cdot\tau, \qquad \tau = \frac{F}{S}, \tag{5}$$

where V_{sh} is the volume of the wall, V the inner volume, S the inner surface, F the force in the muscle cell and τ stress in the muscle fibre, which can be derived as

$$\tau = \frac{\frac{-q}{3\kappa(x\cdot y)^2} + \left[k_1 zy(x-1) + \frac{zyx}{2k_2}(x-1)^2 - \frac{xy}{k_2}C'\right]}{k_1 y + \frac{x^2 y}{k_2}}. \tag{6}$$

This will be putted into the equations for the isotonic contraction.

4 Urethra flow

We now briefly introduce a problem describing fluid flow through the elastic tube. In the case of the male urethra, the system has the following form

$$\begin{aligned} a_t + q_x &= 0, \\ q_t + \left(\frac{q^2}{a} + \frac{a^2}{2\rho\beta}\right)_x &= \frac{a}{\rho}\left(\frac{a_0}{\beta}\right)_x + \frac{a^2}{2\rho\beta^2}\beta_x - \frac{q^2}{4a^2}\sqrt{\frac{\pi}{a}}\lambda(Re), \end{aligned} \tag{7}$$

where $a = a(x,t)$ is the unknown cross-section area, $q = q(x,t)$ is the unknown flow rate, ρ is the fluid density, $a_0 = a_0(x)$ is the cross-section of the tube under no pressure, $\beta = \beta(x,t)$ is the coefficient describing tube compliance and $\lambda(Re)$ is the Mooney-Darcy friction factor ($\lambda(Re) = 64/Re$ for laminar flow). Re is the Reynolds number. This model contains constitutive relation between the pressure and the cross section of the tube

$$p = \frac{a - a_0}{\beta} + p_e, \tag{8}$$

where p_e is surrounding pressure. Presented system (7) can be written in the matrix form

$$\mathbf{u}_t + [\mathbf{f}(\mathbf{u}, x)]_x = \boldsymbol{\psi}(\mathbf{u}, x), \tag{9}$$

with $\mathbf{u}(x,t)$ being the vector of conserved quantities, $\mathbf{f}(\mathbf{u}, x)$ the flux function and $\boldsymbol{\psi}(\mathbf{u}, x)$ the source term. This relation represents the balance laws. For the following consideration, we reformulate this problem to the nonconservative form.

4.1 Decompositions based on augmented system

The numerical scheme for solving problems (9) can be written in fluctuation form

$$\frac{\partial \mathbf{U_j}}{\partial t} = -\frac{1}{\Delta x}[\mathbf{A}^-(\mathbf{U}^-_{j+1/2}, \mathbf{U}^+_{j+1/2}) + \mathbf{A}(\mathbf{U}^-_{j+1/2}, \mathbf{U}^+_{j-1/2}) + \mathbf{A}^+(\mathbf{U}^-_{j-1/2}, \mathbf{U}^+_{j-1/2})], \tag{10}$$

where $\mathbf{A}^{\pm}(\mathbf{U}^-_{j+1/2}, \mathbf{U}^+_{j+1/2})$ are so called fluctuations. They can be defined by the sum of waves moving to the right or to the left. We use the notation $\mathbf{U}^+_{j+1/2}$ and $\mathbf{U}^-_{j+1/2}$ for the approximations of limit values of reconstructions from the discrete cell averages at the points $x_{j+1/2}$. The most common choices are based on the minmod function or ENO and WENO techniques.

The our approach is based on the extension of the system (7) by other equations appropriately chosen degenerate conservation laws. The advantage of this step is in the conversion of the nonhomogeneous system to the homogeneous quasilinear one $\mathbf{w}_t + \mathbf{B}(\mathbf{w})\mathbf{w}_x = \mathbf{0}$ and possibility of preserving general steady states (see [1]). It is very important to choose such approximation which conserves steady states, if these states occur exactly. The steady state for the augmented system means $\mathbf{B}(\mathbf{w})\mathbf{w}_x = \mathbf{0}$, therefore $\mathbf{w}_x$ is a linear combination of the eigenvectors corresponding to the zero eigenvalues.

5 Complex model of the bladder and the urethra

The whole voiding model consists of the detrusor smooth muscle cell model and the model of the urethra flow. It is described by the system of 12 equations describing the bladder model and the detrusor contraction during voiding (1), (2) and (4) and $2J$ equations of urethra flow, where

J is the number of finite volumes of the urethra region. The connection between the detrusor model and urethra flow is implemented by the relation (6) and the constitutive relation (8). The outflow of the bladder is the same as the inflow to the urethra region. So the pressure of the bladder is dependent on the flow rate in the tube (6). The cross-section in the first finite volume of the urethra region is then given by the constitutive relation (8). From the view of urethra flow, the inflow boundary condition consists of the given cross-section and extrapolation of the flow rate from the urethra region.

6 Conclusion

We presented the complex model of the lower part of the urinary tract. A simple bladder model and the detrusor contraction model were developed during voiding together with the detailed model of urethra flow. The urethra flow was described by the high-resolution positive semidefiniteness method, which preserves general steady states.

Acknowledgement: This work was supported by the European Regional Development Fund (ERDF), project NTIS - New Technologies for Information Society, European Centre of Excellence, CZ.1.05/1.1.00/02.0090 and the project SGS-2010-077 Support of Biomechanics at the Faculty of Applied Sciences, University of West Bohemia in Pilsen.

References

[1] M. Brandner, J. Egermaier, H. Kopincová: *Augmented Riemann solver for urethra flow modelling.* In: Mathematics and Computers in Simulations 80 (6), 2009, pp. 1222–1231.

[2] A. Dicarlo, S. Quiligotti: *Growth and balance.* In: Mechanics Research Communications 29, Pergamon Press, 2002, pp. 449–456.

[3] C.M. Hai, R.A. Murphy: *Adenosine 5'-triphosphate consumption by smooth muscle as predicted by the coupled four-state crossbridge model.* In: Biophysical Journal 61 (2), 1992, pp. 530–541.

[4] M. Koenigsberger, R. Sauser, D. Seppey, J.-L. Beny, J.-J., Meister: *Calcium dynamics and vosomotion in arteries subject to isometric, isobaric and isotonic conditions.* In: Biophysical Journal, 95, 2008, pp. 2728–2738.

[5] J. Laforet, D. Guiraud: *Smooth muscle model for functional electric stimulation applications.* In: Proceedings of the 29th Annual International Conference of the IEEE EMBS, Cite Internationale, Lyon. France August, 2007, pp. 23–26.

[6] J. Rosenberg, L. Hynčík: *Modelling of the influance of the stifness evolution on the behaviour of the Muscle fibre.* In: Human Biomechanice 2008, International Conference, 29.9.–1.10.2008 Praha, Czech Republic.

[7] J. Rosenberg, M. Svobodová: *Comments on the thermodynamical background to the growth and remodelling theory applied to the model of muscle fibre contraction.* In: Applied and Computational Mechanics 4 (1), 2010, pp. 101–112.

Analytical solution for singularities in Stokes flow

P. Burda, J. Novotný, J. Šístek

[1] Czech Technical University in Prague and VŠB Technical University of Ostrava
[2] Department of Mathematics, Czech Technical University in Prague
[3] Institute of Mathematics AS CR, Prague

1 Introduction

The behaviour of the solution of Stokes and Navier-Stokes equations in domains with corners or with discontinuities in boundary conditions is still not quite well understood. We use the analytical solution to characterize the singular part of the solution. The asymptotics apply also to Navier-Stokes equations. The results are applied to two examples: the flow in a channel with forward and backward steps, and the problem of lid driven cavity.

2 Analytical solution of the Stokes flow near corners

We consider the Stokes problem in vorticity - stream function formulation, cf [1], and transform the problem to polar coordinates $x = r\cos\vartheta$, $y = r\sin\vartheta$, with the pole in the corner P, cf. Fig. 1.

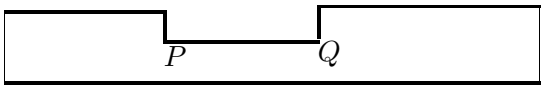


Figure 1: The solution domain Ω.

So we have to find stream function $\psi(r,\vartheta)$ and vorticity $\omega(r,\vartheta)$, satisfying the equations

$$\frac{\partial^2\psi}{\partial r^2}+\frac{1}{r}\frac{\partial\psi}{\partial r}+\frac{1}{r^2}\frac{\partial^2\psi}{\partial\vartheta^2} \;=\; -\omega\ , \qquad \frac{\partial^2\omega}{\partial r^2}+\frac{1}{r}\frac{\partial\omega}{\partial r}+\frac{1}{r^2}\frac{\partial^2\omega}{\partial\vartheta^2} \;=\; 0\ . \tag{1}$$

To solve the equations (1) we use separation of variables,

$$\psi(r,\vartheta)=P(r)\cdot F(\vartheta)\ , \qquad \omega(r,\vartheta)=R(r)\cdot G(\vartheta)\ . \tag{2}$$

Analyzing arizing differential equations we come to the asymptotic formula for stream function

$$\psi(r,\vartheta)=r^{-\sqrt{\kappa}+2}\cdot F(\vartheta) \quad (+h.o.t.), \tag{3}$$

where κ is a positive parameter depending only on the angle of the corner.

Example 1. We consider flow in 2D region with boundary corner of internal angle φ, as e.g. on Fig. 1. We assume nonslip boundary conditions, so the boundary conditions for the stream function are

$$\psi(r,0)=0,\ \ \psi(r,\varphi)=0, \qquad \frac{\partial\psi}{\partial\vartheta}(r,0)=0, \qquad \frac{\partial\psi}{\partial\vartheta}(r,\varphi)=0. \tag{4}$$

As an example we take the domain shown in Fig. 1, where the angle $\varphi = \frac{3}{2}\pi$. Then we get $\sqrt{\kappa} = 0.45552$. Now, by (3) we get e.g. the asymptotics for stream function, near the corner P

$$\psi(r, \vartheta) = r^{1.54448} \cdot F(\vartheta) \quad (+h.o.t.), \tag{5}$$

with F independent of r. So we get the asymptotics for velocity components and pressure

$$u_r = r^{0.54448}\ F_1(\vartheta), \quad u_\vartheta = r^{0.54448}\ F_2(\vartheta), \quad p = r^{-0.45552}\ F_3(\vartheta), \tag{6}$$

where $F_1(\vartheta), F_2(\vartheta), F_3(\vartheta)$ are independent of r. The same formulas apply to point Q.

Example 2.

Let us consider 2D flow in lid driven cavity, see Fig. 2, with boundary conditions

$$\psi(r, \frac{3}{2}\pi) = 0, \quad \psi(r, 2\pi) = 0, \tag{7}$$

$$\frac{1}{r}\frac{\partial \psi}{\partial \vartheta}(r, \frac{3}{2}\pi) = 0, \quad \frac{1}{r}\frac{\partial \psi}{\partial \vartheta}(r, 2\pi) = 1, \tag{8}$$

for left upper corner.

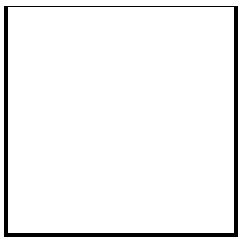


Figure 2: The lid driven cavity.

We solve the equations (1) similarly as above, by means of separation (2). One can then derive the asymptotics in upper corners of the cavity

$$\psi(r, \vartheta) = r \cdot F(\vartheta), \quad u_r = F'(\vartheta), \quad u_\vartheta = F(\vartheta), \quad p(r, \vartheta) = \frac{1}{r}\Phi(\vartheta), \tag{9}$$

which are much worse than in case of the corner in Example 1.

3 Application to finite element calculations

The application of the asymptotics may be at least twofold. First, the analytical solution near corners may be used to direct checking of numerical solution. Second, combining the asymptotics of Navier-Stokes equations with a priori estimates we get an algorithm for generating the finite element mesh at such corners cf. [2, 3]. As an application we show on Fig. 3 the locally refined mesh near upper corners of lid driven cavity, and pressure calculated by this algorithm.

4 Conclusion

We solve analytically the Stokes problem in 2D domains, using polar coordinates and separation of variables. This is then used to find the asymptotics of the solution near corners, also for Navier-Stokes equations. We show application to very precise finite element solution.

Acknowledgement. This work has been supported by the grant No. 106/08/0403 - GACR and by the project IT4Innovations.

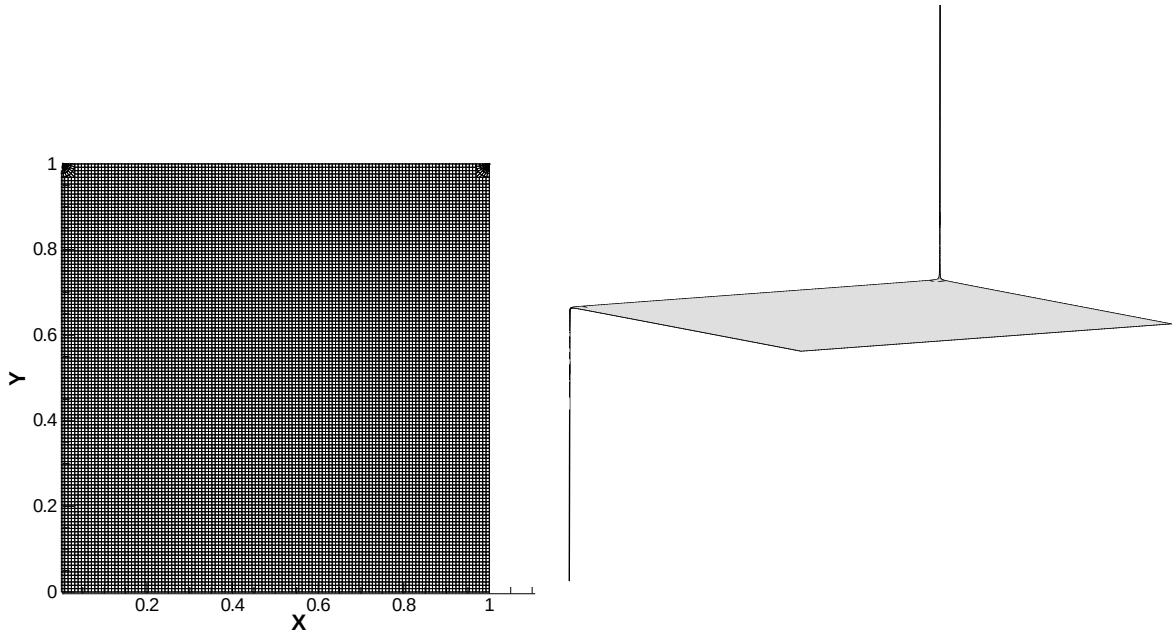


Figure 3: Lid driven cavity, Re = 10,000 Left: mesh 128×128 refined locally Right: pressure.

References

[1] G. K. Batchelor: *An introduction to fluid dynamics*. Cambridge University Press, 1967.

[2] P. Burda, J. Novotný, J. Šístek: *Precise FEM solution of a corner singularity using an adjusted mesh.* Int. J. Numer. Meth. Fluids 47, 2005, pp. 1285–1292.

[3] P. Burda, J. Novotný, J. Šístek: *Singularities in lid driven cavity solved by adjusted finite element method*. In: Computational Fluid Dynamics 2011, A. Kuzmin (ed.), Springer-Verlag, 2011, pp. 799–805.

Numerical solution of perfect plastic problems with contact: part II – implementation

M. Čermák, S. Sysala, J. Haslinger

IT4Innovations, VŠB - Technical University of Ostrava
Institute of Geonics AS CR, Ostrava
Charles University, Prague

1 Introduction

Our contribution is divided into two parts. In Part I, see [6], we focus on theory of discretized problems and suitable numerical methods. In Part II, we describe implementation of the problem and illustrate it on a model example.

In Part I [6], we have proposed the modified semismooth Newton method for the primal formulation of the problem and mentioned the Uzawa method for the augmented Lagrangian formulation of the problem. In each step of both methods, we mainly solve a problem that is similar to the contact problem with elastic bodies. This inner problem can be classified as a quadratic problem with simple constraints.

In Part II, we rewrite the inner problem on its dual form in terms of Lagrange multipliers enforcing the non-penetration condition on the contact zones. The dual problem is solved by the SMALSE method [3]. For a parallel implementation, we combine the method with the TFETI domain decomposition method [2], see Section 2. The whole contact problems of elastic-perfectly plastic bodies is implemented in MatLab within the MatSol library [5]. We illustrate the investigated numerical methods introduced in Part I [6] and Part II in Section 3.

2 Solution of the inner problem

Since we apply the TFETI domain decomposition method [2], we tear the bodies from the parts of the boundaries with the Dirichlet boundary condition, decompose it into subdomains, assign each subdomain by a unique number, and introduce new "gluing" conditions on the artificial intersubdomain boundaries and on the boundaries with imposed Dirichlet condition. In particular, the domain $\Omega_h^i \equiv \Omega^i$ is decomposed into a system of s_i disjoint polynomial subdomains $\Omega^{i,p} \subset \Omega^i$, $p = 1, 2, \ldots, s_i$, $i = 1, 2$, see Fig. 1. The partition corresponds to the finite element partition described in Part I [6].

We introduce an algebraic scheme of the inner problem related to the domain decomposition. It means that a displacement vector $\mathbf{v} \in \mathbb{R}^n$ has the following structure:

$$\mathbf{v} = \left(\mathbf{v}_{1,1}^T, \mathbf{v}_{1,2}^T, \ldots, \mathbf{v}_{1,s_1}^T, \mathbf{v}_{2,1}^T, \ldots, \mathbf{v}_{2,s_2}^T\right)^T,$$

where $\mathbf{v}_{i,p}$ denotes the displacement vector on $\Omega^{i,p}$, $i = 1, 2$. Then the algebraic representations of the space V and the set K introduced in Part I [6] are defined as follows:

$$\mathcal{V} := \{\mathbf{v} \in \mathbb{R}^n \mid \mathbf{B}_E\mathbf{v} = \mathbf{o}\}, \tag{1}$$

$$\mathcal{K} := \{\mathbf{v} \in \mathbb{R}^n \mid \mathbf{B}_E\mathbf{v} = \mathbf{o},\ \mathbf{B}_I\mathbf{v} \leq \mathbf{c}_I\}. \tag{2}$$

Here the equality constraint matrix $\mathbf{B}_E \in \mathbb{R}^{m_E \times n}$ represents the gluing conditions among neighbouring subdomains and the Dirichlet boundary conditions. The inequality constraint matrix $\mathbf{B}_I \in \mathbb{R}^{m_I \times n}$ represents the non-penetration condition on the contact zones.

Let $\mathbf{K}_e \in \mathbb{R}^{n \times n}$ be a block diagonal matrix consisting of the elastic stiffness matrices $\mathbf{K}_e^{i,p}$ defined on each subdomain $\Omega^{i,p}$, $i = 1, 2$, $p = 1, \ldots, s_i$. Due to the presence of the Dirichlet boundary conditions on both subdomains and the Korn inequality, we can define the energy norm on $\mathcal{V}$:

$$\|\mathbf{v}\|_e := \sqrt{\mathbf{v}^T \mathbf{K}_e \mathbf{v}} = \sqrt{\sum_{i=1}^{2} \sum_{p=1}^{s_i} \mathbf{v}_{i,p}^T \mathbf{K}_e^{i,p} \mathbf{v}_{i,p}}, \quad \mathbf{v} = \left(\mathbf{v}_{1,1}^T, \ldots, \mathbf{v}_{1,s_1}^T, \mathbf{v}_{2,1}^T, \ldots, \mathbf{v}_{2,s_2}^T\right)^T \in \mathcal{V}.$$

The scheme of the inner problem is the following:

$$\text{find } \mathbf{u} \in \mathcal{K}_k : \quad J_k(\mathbf{u}) \leq J_k(\mathbf{v}) \qquad \forall \mathbf{v} \in \mathcal{K}_k, \tag{3}$$

where

$$J_k(\mathbf{v}) := \frac{1}{2} \mathbf{v}^T \mathbf{K}_k \mathbf{v} - \mathbf{f}_k^T \mathbf{v}, \quad \mathbf{v} \in \mathcal{K}_k. \tag{4}$$

Here k denotes the k-step of both methods. In case of the Uzawa method $\mathcal{K}_k = \mathcal{K}$, $\mathbf{K}_k = \mathbf{K}_e$ and $\mathbf{u}$ represent the displacement at the next step $k+1$. In case of the Newton method,

$$\mathcal{K}_k := \mathcal{K} - \mathbf{u}^k = \{\mathbf{v} \in \mathbb{R}^n \ ; \ \mathbf{B}_E \mathbf{v} = \mathbf{o}, \ \mathbf{B}_I \mathbf{v} \leq \mathbf{c}_{I,k}\}, \quad \mathbf{c}_{I,k} = \mathbf{c}_I - \mathbf{B}_I \mathbf{u}^k,$$

$\mathbf{K}_k$ represents the function $T^{o,\nu}$ introduced in Part I [6], i.e.

$$\nu \|\mathbf{w}\|_e^2 \leq \mathbf{w}^T \mathbf{K}_k \mathbf{w} \leq \|\mathbf{w}\|_e^2 \quad \forall \mathbf{v}, \mathbf{w} \in \mathcal{V}, \tag{5}$$

and $\mathbf{u}$, $\mathbf{u}^k$ represent δu^k, u^k from Part I [6], respectively. For both methods $\mathbf{f}_k$ denotes the load vector in dependence on the k-th step. The problem (3) is practically the same as contact problems of elastic bodies. Therefore we can use the same techniques as in [4], [3] or [1] based on the SMALSE method.

To use the method, we replace all the constraints by the Lagrange multipliers, see the Figure 1. In particular, we use two types of Lagrange multipliers, namely $\boldsymbol{\lambda}_I \in \mathbb{R}^{m_I}$, $\boldsymbol{\lambda}_I \geq \mathbf{o}$ related to the non-penetration condition, $\boldsymbol{\lambda}_E \in \mathbb{R}^{m_E}$ related to the "gluing" and Dirichlet conditions. To simplify the notation, we denote

$$\boldsymbol{\lambda} = \begin{bmatrix} \boldsymbol{\lambda}_E \\ \boldsymbol{\lambda}_I \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{B}_E \\ \mathbf{B}_I \end{bmatrix}, \quad \mathbf{c}_k = \begin{bmatrix} \mathbf{o} \\ \mathbf{c}_{I,k} \end{bmatrix},$$

and

$$\Lambda = \{\boldsymbol{\lambda} = (\boldsymbol{\lambda}_E^T, \boldsymbol{\lambda}_I^T)^T \in \mathbb{R}^{m_E + m_I} : \ \boldsymbol{\lambda}_I \geq \mathbf{o}\}.$$

Then the Lagrangian associated with problem (3) reads as

$$L_k(\mathbf{v}, \boldsymbol{\lambda}) = \frac{1}{2} \mathbf{v}^T \mathbf{K}_k \mathbf{v} - \mathbf{f}_k^T \mathbf{v} + \boldsymbol{\lambda}^T (\mathbf{B}\mathbf{v} - \mathbf{c}_k), \quad \mathbf{v} \in \mathbb{R}^n, \ \boldsymbol{\lambda} \in \Lambda. \tag{6}$$

Using the convexity of the cost function and constraints, we can use the classical duality theory to reformulate problem (3) to get

$$J_k(\mathbf{u}) = \min_{\mathbf{v} \in \mathcal{K}_k} J_k(\mathbf{v}) = \min_{\mathbf{v} \in \mathbb{R}^n} \sup_{\boldsymbol{\lambda} \in \Lambda} L_k(\mathbf{v}, \boldsymbol{\lambda}) = \max_{\boldsymbol{\lambda} \in \Lambda} \inf_{\mathbf{v} \in \mathbb{R}^n} L_k(\mathbf{v}, \boldsymbol{\lambda}) = \max_{\boldsymbol{\lambda} \in \Lambda} \{-\boldsymbol{\Theta}_k(\boldsymbol{\lambda})\}, \tag{7}$$

with

$$\boldsymbol{\Theta}_k(\boldsymbol{\lambda}) = \begin{cases} \frac{1}{2} \boldsymbol{\lambda}^T \mathbf{B} \mathbf{K}_k^\dagger \mathbf{B}^T \boldsymbol{\lambda} - \boldsymbol{\lambda}^T (\mathbf{B} \mathbf{K}_k^\dagger \mathbf{f}_k - \mathbf{c}_k), & \mathbf{R}_k^T (\mathbf{f}_k - \mathbf{B}^T \boldsymbol{\lambda}) = \mathbf{o}, \\ +\infty, & \text{otherwise}, \end{cases}$$

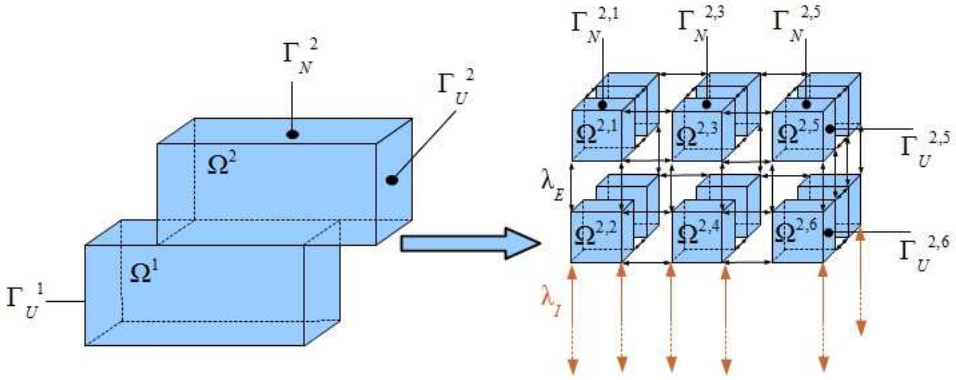


Figure 1: Scheme of the geometry and domain decomposition.

where $\mathbf{K}_k^\dagger$ is a pseudoinverse matrix to $\mathbf{K}_k$ and $\mathbf{R}_k \in \mathbb{R}^{n\times l}$ represents the null space of $\mathbf{K}_k$. Thus the corresponding dual problem has the form:

$$\text{find } \boldsymbol{\lambda}^{\boldsymbol{k}} \in \Lambda: \quad \boldsymbol{\Theta}_k(\boldsymbol{\lambda}^k) \le \boldsymbol{\Theta}_k(\boldsymbol{\lambda}) \quad \forall \boldsymbol{\lambda} \in \Lambda. \tag{8}$$

We solve the dual problem by algorithm SMALSE-M [3]. The algorithm is based on active set strategy and it combines three steps: CG with preconditioning based on orthogonal projectors, expansion, and proportioning.

Once the solution $\boldsymbol{\lambda}^k$ of (8) is known, the solution of (3) can be evaluated in this way:

$$\mathbf{u} = \mathbf{K}_k^\dagger(\mathbf{f} - \mathbf{B}^T\boldsymbol{\lambda}^k) + \mathbf{R}_k\boldsymbol{\alpha}_k, \quad \boldsymbol{\alpha}_k = (\mathbf{R}_k^T\overline{\mathbf{B}}^T\overline{\mathbf{B}}\mathbf{R}_k)^{-1}\mathbf{R}_k^T\overline{\mathbf{B}}^T(\overline{\mathbf{c}}_k - \overline{\mathbf{B}}\mathbf{K}_k^\dagger(\mathbf{f}_k - \mathbf{B}^T\boldsymbol{\lambda}^k)),$$

where the matrix $\overline{\mathbf{B}}$ and the vector $\overline{\mathbf{c}}_k$ are formed by the rows of $\mathbf{B}$ and $\mathbf{c}_k$ corresponding to all equality constraints and all active inequality constraints.

Notice that we use in fact the inexact Newton method with respect to computing of $\mathbf{u}$.

3 Numerical experiment

In this section we compare numerical methods introduced in Part I [6] on a numerical example. The geometry of the problem is depicted in Figure 1. The dimensions of Ω^1, Ω^2 are $3000\times1000\times1000$. The indicated traction forces are prescribed by the constant function $g = 150$. The mesh is generated in MatSol and has 53 802 nodes and 288 000 tetrahedrons. Finally, we decompose Ω^1, Ω^2 into 48 subdomains. After decomposition we have 191 664 primal variables, 33 933 dual variables, and from these are 1 029 contact pairs. The bodies Ω^1, Ω^2 are made of homogenous isotropic materials with the parameters $E^1 = E^2 = 206\,900$, $\nu^1 = \nu^2 = 0.29$, and $\sigma_y^1 = \sigma_y^2 = 450$. The influence of the loading parameter λ has not been investigated yet in this example, i.e. we set $\lambda = 1$. The proposed algorithms are parallelized using Matlab Distributed Computing Server and Matlab Parallel Toolbox. For all computations we use 24 cores with 2GB memory per core of the HP Blade system, model BLc7000. Since we expect that the choisen load is far from the limit load, we use the stopping criterion which compare relative displacement increments.

In Table 1, we see the iteration process of the Newton method for $\nu = 0$. In this table we show, how the program behaves in each Newton iteration. The column "Val. of $\mathcal{J}_\lambda$" means the value of the nonlinear functional $\mathcal{J}_\lambda$ defined in Part I.

Numb. of Newt. its.	SMALSE its.	Hessian multipl.	Numb. of plas. els.	Conv. disp.	Time	Val. of $\mathcal{J}_\lambda$ times 10^6
1	21	272	0	1	82.2	-1 011.622
2	10	656	22 246	4.5328e-1	147.3	-1 102.614
3	9	829	32 624	1.2649e-1	200.9	-1 130.835
4	8	1 317	38 918	6.2349e-2	323.3	-1 136.048
5	8	1 399	39 994	2.8553e-2	274.1	-1 139.456
6	8	1 406	41 055	2.1786e-2	301.3	-1 139.540
7	8	1 564	41 236	2.8527e-3	343.3	-1 139.540
8	8	1 618	41 236	1.5401e-6	383.2	-1 139.540

Table 1: The Newton method for $\nu = 0$.

ν	Numb. of Newt. its.	SMALSE-M its.	Hessian multi.	Numb. of plas. els.	Time for 1 New. it.	total time
0	8	8	1 618	41 236	383.2	2 067.5
0.05	24	13	743	41 236	177.1	3 757.7
0.10	33	15	572	41 216	140.9	4 551.7
0.15	45	16	507	41 220	136.4	5 644.8
0.20	54	17	472	41 209	126.2	6 383.4
0.25	62	17	428	41 200	114.5	6 803.1
0.30	69	18	332	41 181	107.5	7 042.0
1.00	139	22	275	40 886	98.1	14 099.7

Table 2: The Newton method with approx hessian by parametr ν.

r	Numb. of iters.	SMALSE-M iters.	Hessian multi.	Numb. of plas. els.	Time for 1 iter.	total time
0.05	86	20	395	41 192	102.0	8 178.4
0.10	58	22	279	41 200	74.8	4 618.8
0.15	47	22	273	41 216	72.1	3 992.1
0.20	45	22	276	41 223	75.8	3 790.6
0.25	54	22	276	41 218	73.6	4 446.9
0.30	63	22	275	41 212	72.3	5 297.8
0.35	71	22	277	41 209	75.4	5 923.3
0.40	78	22	276	41 206	73.4	6 423.9

Table 3: The Uzawa algorithm with parametr r.

In Table 2, we compare the number of Newton iteration, the average number of SMALSE-M iteration for one Newton iteration, the average number of Hessian multiplication, the worst time for one Newton iteration and the total time for the Newton method in dependence on the regularization parameter ν. In this case, we observe the best convergence for $\nu = 0$. Therefore we suppose that the prescribed load is far from the limit load based on the theoretical results from Part I.

In Table 3, we compare similar quantities as in Table 2 for the Uzawa algorithm in dependence on the penalty parameter $r > 0$ from the augmented Lagrangian formulation of the problem. We see that the best convergence results are observed for $r = 0.2$. However it seems to be problematic to estimate an optimal value of r a priori.

In Figures 2 and 3, there are depicted the von Mises stress distribution and total displacement which are the same for both methods.

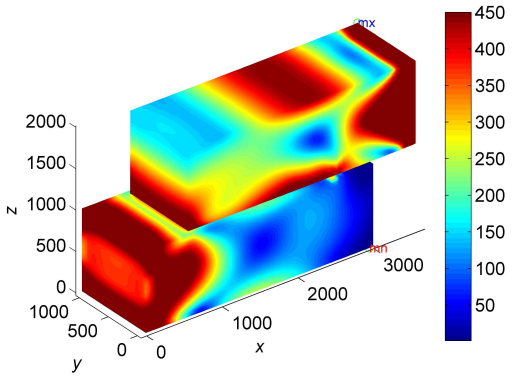


Figure 2: Distribution of von Mises stress.

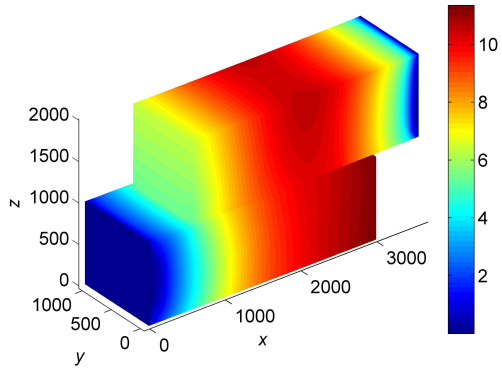


Figure 3: Total displacement.

4 Conclusion

In this contribution, we have described some implementation details of the contact problems of elastic-perfectly plastic bodies. We have also illustrated the Newton and Uzawa methods on the numerical example. We plan to study stability and robustness of the methods in dependence on increasing λ up to the limit load. We also plan to use different numerical methods for solving the inner problem like the semi-smooth Newton method for its primal-dual formulation.

Acknowledgement: This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and by the project SPOMECH - Creating a multidisciplinary R&D team for reliable solution of mechanical problems, reg. no. CZ.1.07/2.3.00/20.0070 within Operational Programme 'Education for competitiveness' funded by Structural Funds of the European Union and state budget of the Czech Republic.

References

[1] M. Čermák: *Scalable algorithms for solving elasto-plastic problems*. Doctoral thesis in VSB-TU Ostrava, 2012.

[2] Z. Dostál, D. Horák, R. Kučera: *Total FETI – an easier implementable variant of the FETI method for numerical solution of elliptic PDE*. Communications in Numerical Methods in Engineering 22 (12), 2006, pp. 1155–1162.

[3] Z. Dostál, T. Kozubek: *An optimal algorithm and superrelaxation for minimization of a quadratic function subject to separable convex constraints with applications*. Mathematical Programming 135 (1–2), 2012, pp. 195–220.

[4] Z. Dostál, T. Kozubek, A. Markopoulos, T. Brzobohatý, V. Vondrák, P. Horyl: *Theoretically supported scalable TFETI algorithm for the solution of multibody 3D contact problems with friction*. CMAME 205–208, 2012, pp. 110–120.

[5] T. Kozubek, A. Markopoulos, T. Brzobohatý, R. Kučera, V. Vondrák, Z. Dostál: *MatSol – MATLAB efficient solvers for problems in engineering*. `http://matsol.vsb.cz/`.

[6] S. Sysala, J. Haslinger, M. Cermak: *Numerical solution of perfect plastic problems with contact: part I – theory and numerical methods*. In proceedings of Seminar on Numerical Analysis, 2013.

On using unitary matrices for the investigation of GMRES convergence behavior

J. Duintjer Tebbens

Institute of Computer Science AS CR, Prague

1 Introduction

In this extended abstract, we consider the convergence behavior of the GMRES method [9] for solving linear systems

$$Ax = b, \qquad A \in C^{n\times n},\, b \in C^n.$$

With zero initial guess $x_0 = 0$, the kth GMRES iterate is the vector x_k in the kth Krylov subspace minimizing the residual norm, i.e.

$$x_k = \arg \min_{x\in\mathcal{K}_k(A,b)} \|b - Ax\|, \qquad \mathcal{K}_k(A,b) \equiv \mathrm{span}\{b,\, Ab,\, \ldots,\, A^{k-1}b\}. \tag{1}$$

Hence the kth residual vector $r_k = b - Ax_k$ is the difference between b and its orthogonal projection onto the Krylov *residual* subspace $A\mathcal{K}_k(A,b)$.

It has been known for some time that eigenvalues alone cannot explain GMRES convergence behavior for general non-normal matrices. This was shown in the 1994 paper [5], in which the authors studied so-called GMRES(A,b)-equivalent matrices. A GMRES(A,b)-equivalent matrix B generates the same Krylov residual space as the one given by the pair (A,b), that is

$$B\mathcal{K}_k(B,b) = A\mathcal{K}_k(A,b),\ k = 1,2,\ldots,n$$

(we assume throughout, that GMRES applied to A, b does not terminate until the step n, i.e., $\dim(\mathcal{K}_n(A,b)) = n$). Then GMRES applied to (B,b) yields the same convergence history (with respect to residual norms) as GMRES applied to (A,b). It was proved in [5] that the spectrum of B can consist of arbitrary nonzero values. In [6] this was complemented with the fact that any nonincreasing sequence of residual norms can be generated by GMRES and [1] closed this series of papers with a description of the class of matrices and right-hand sides giving prescribed convergence history while the system matrix has prescribed nonzero spectrum; for a survey see [7, Section 5.7]. In [2] one finds a parametrization of the class of matrices and right-hand sides generating, in addition to prescribed residual norms and eigenvalues, prescribed Ritz values in all iterations.

All these results show that spectral information can be very misleading when explaining GMRES convergence behavior with general, non-normal matrices. On the other hand, for *normal* matrices the behavior of the GMRES method is well-understood in terms of the eigenvalues of the matrix and the components of the right-hand side in the eigenvector basis. It was shown in [5] that for every pair (A,b) there always exist GMRES(A,b)-equivalent matrices B which are normal and even unitary. Therefore we can try to analyze the behavior of the GMRES method applied to (A,b) with the spectral properties of any normal GMRES(A,b)-equivalent matrix. The goal of this extended abstract is to explain how the eigenvalues of a unitary GMRES(A,b)-equivalent matrix are related to properties of the pair (A,b) and to briefly discuss what these eigenvalues can tell about the convergence of GMRES applied to (A,b). For proofs and more details on the presented material, see the forthcoming publication [4]. This is joint work with Gérard Meurant and Zdeněk Strakoš.

2 Eigenvalues of unitary GMRES(A, b)-equivalent matrices

Unitary GMRES(A, b)-equivalent matrices can be characterized as follows [4].

Theorem 2.1. *Let $A \in C^{n\times n}$ be nonsingular and let $b \in C^n$. The following assertions are equivalent:*

1- B is unitary and GMRES(A, b)-equivalent,

2- $B = WV^*$, where V is a unitary matrix whose first k columns give a basis of $\mathcal{K}_k(A, b)$ for $1 \le k \le n$ and W is a unitary matrix whose first k columns give a basis of $A\mathcal{K}_k(A, b)$ for $1 \le k \le n$.

It follows that the eigenvalues of unitary GMRES (A, b)-equivalent matrices are the eigenvalues of generalized eigenvalue problems of the form

$$V^*x = \mu W^*x,$$

where V and W are as defined in the previous theorem. The same holds for the eigenvalues of unitary matrices C such that the pair (C, c), with c not necessarily equal to b, generates the same GMRES convergence curve as (A, b) [4]. Note that V and W depend strongly on the interplay between A and b, hence the eigenvalues μ will in general also depend on this interplay and not on properties of A alone.

It is clear from Theorem 2.1 that there may exist unitary GMRES(A, b)-equivalent matrices with different spectra: If V is a unitary matrix whose first k columns give a basis of $\mathcal{K}_k(A, b)$ for $1 \le k \le n$, then so is VD^* for any diagonal unitary matrix D. Hence all matrices of the form

$$WDV^*, \qquad D \quad \text{is diagonal and unitary,}$$

are unitary GMRES(A, b)-equivalent. But the spectra of WV^* and WDV^* can differ significantly; they need not be rotations the one of the other and they do not interlace in general.

Let us give a small example. We can construct an unreduced upper Hessenberg matrix H of size seven such that GMRES applied to (H, e_1), e_1 being the first column of the identity, generates the residual norms

$$\begin{aligned} \|r_1\| &= 0.5, \quad \|r_2\| = 0.1, \\ \|r_3\| &= 0.05, \quad \|r_4\| = 0.01, \\ \|r_5\| &= 0.005, \quad \|r_6\| = 0.001. \end{aligned} \tag{2}$$

To achieve this, we can use the parametrization of [3, Theorem 2] and define H as

$$H = U^{-1}CU, \quad U = \begin{bmatrix} & g^T \\ 0 & T \end{bmatrix},$$

with

$$g_1 = 1, \qquad g_k = \frac{\sqrt{\|r_{k-2}\|^2 - \|r_{k-1}\|^2}}{\|r_{k-2}\|\|r_{k-1}\|}, \qquad k = 2, \ldots, 7$$

and where C is the companion matrix of a polynomial having as its roots the eigenvalues of H. Here we choose $T = I_6$ and we choose the spectrum of H to consist of the value 1. A unitary matrix V whose first k columns give a basis of $\mathcal{K}_k(H, e_1)$ for $1 \le k \le n$ is given by $V = I_7$ and a unitary matrix W whose first k columns give a basis of $H\mathcal{K}_k(H, e_1)$ for $1 \le k \le n$ is

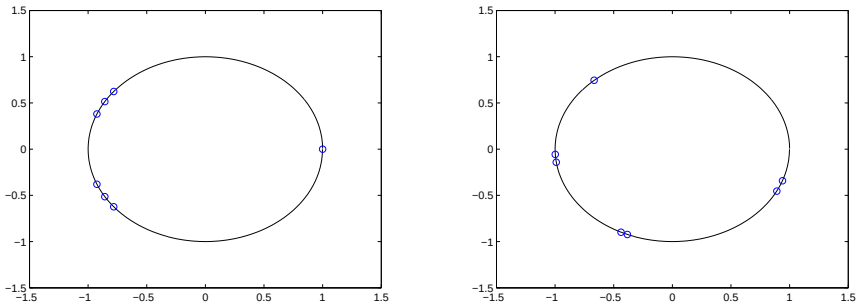


Figure 1: Spectrum of Q (left) and of QD_1 (right).

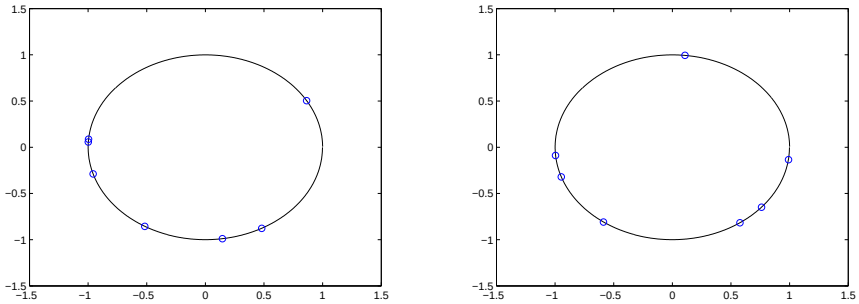


Figure 2: Spectrum of QD_2 (left) and of QD_3 (right).

any Q factor Q of a QR factorization of H. Hence with Theorem 2.1, $B = WV^* = Q$ is GMRES(H, e_1)-equivalent and so is QD for any diagonal unitary D. We computed a Q factor Q of a QR factorization of H and the spectrum of this (real) Q is displayed on the left part of Figure 1. The spectra of QD_1, QD_2 and QD_3 where $D_i, i = 1, 2, 3$ are random (complex) diagonal unitary matrices, are displayed on the right part of Figure 1 and in Figure 2. The four spectra do not seem to be related by any special properties, but GMRES applied to (QD_i, e_1) and to (Q, e_1) generates the residual norm history (2) for all $i = 1, 2, 3$.

Thus, in general there will be more than one unitary spectrum corresponding to a certain GMRES convergence curve and one may ask wether the eigenvalues of unitary GMRES(A, b)-equivalent matrices need tell us anything at all about GMRES-convergence for (A, b). In fact, all we now is that if the spectrum of a unitary equivalent matrix has a large maximum gap, then we have fast GMRES convergence. This was shown in [8]. On the other hand, fast convergence can be forced with *any* unitary spectrum by appropriate choice of the right-hand side [4]. A special case is when GMRES stagnates. Then the corresponding unitary spectra will all be rotations of the roots of unity [10]. This result is slightly modified in case of partial stagnation [4]. But as follows from what we mentioned, if a unitary equivalent matrix has a spectrum representing a rotation of the roots of unity, it may also generate fast convergence if we choose the right-hand side appropriately.

Summarizing, in special situations the eigenvalues of unitary GMRES(A, b)-equivalent matrices can tell us something on the convergence of GMRES for (A, b) and vice-versa, some special cases of convergence behavior for (A, b) determine the eigenvalues of unitary GMRES(A, b)-equivalent matrices. In general, however, looking only at the eigenvalues of unitary matrices is not enough to explain GMRES convergence. Components of the right-hand side in the eigenvector basis

must also be taken into account. If time allows it, the last part of the talk will address a novel formula for the kth GMRES residual norm generated with normal matrices, which contains only eigenvalues and components of the right-hand side in the eigenvector basis.

Acknowledgement: The work of J. Duintjer Tebbens is a part of the Institutional Research Plan AV0Z10300504 and it was supported by the project M100301201 of the institutional support of the AS CR.

References

[1] M. Arioli, V. Pták, Z. Strakoš: *Krylov sequences of maximal length and convergence of GMRES*. B.I.T. 38, 1998, pp. 636–643.

[2] J. Duintjer Tebbens, G. Meurant: *Any Ritz value behavior is possible for Arnoldi and for GMRES*. SIMAX 33, 2012, pp. 358–378.

[3] J. Duintjer Tebbens, G. Meurant: *Prescribing the behavior of early terminating GMRES and Arnoldi iterations*. Submitted to Numerical Algorithms in 2012.

[4] J. Duintjer Tebbens, G. Meurant, H. Sadok, Z. Strakoš: *On investigating GMRES convergence using unitary matrices*. In preparation.

[5] A. Greenbaum, Z. Strakoš: *Matrices that generate the same Krylov Residual Spaces*. In Recent Advances in Iterative Methods, G. H. Golub, A. Greenbaum and M. Luskin (eds.), 50, 1994, pp. 95–118.

[6] A. Greenbaum, V. Pták, Z. Strakoš: *Any nonincreasing convergence curve is possible for GMRES*. SIAM J. Matrix Anal. Appl. 17, 1996, pp. 465–469.

[7] J. Liesen, Z. Strakoš: *Krylov subspace methods, principles and analysis*. Oxford University Press, ISBN 978-0-19-965541-0, 2012, 408 pages.

[8] J. Liesen: *Computable convergence bounds for GMRES*. SIAM J. Matrix Anal. Appl. 21, 2000, pp. 882–903.

[9] Y. Saad, M. H. Schultz: *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*. SIAM J. Sci. Stat. Comput. 7, 1986, pp. 856–869.

[10] I. Zavorin, D. P. O'Leary, H. Elman: *Complete stagnation of GMRES*. Linear Algebra Appl. 367, 2003, pp. 165–183.

Composite polynomial convergence bounds, the CSI method and finite precision CG computations

T. Gergelits, Z. Strakoš

Faculty of Mathematics and Physics, Charles University in Prague

1 Introduction

The conjugate gradient method (CG) [6] is used for solving linear algebraic system

$$Ax = b \tag{1}$$

with Hermitian and positive definite (HPD) matrix $A \in \mathbb{C}^{N\times N}$ which is large and sparse. The CG method is *nonlinear* (see, e.g., a thorough discussion in [12]) and it exhibits the so-called superlinear convergence, i.e., it tends to accelerate during computations. The bound most commonly associated with the convergence rate of CG is, however, *linear* and thus unable to describe this phenomenon. In case of isolated large eigenvalues, Axelsson [1] and Jennings [8] describe the CG superlinear convergence behaviour via the so-called composite polynomial bounds. Assuming exact arithmetic, they work quite well. Since the finite precision CG behaviour is *quantitatively* and *qualitatively* different from the CG behaviour in exact arithmetic, the composite polynomial bounds must fail in practical applications. Despite experimental warnings (see, e.g., [8, 17]) and clear theoretical arguments ([5]), misleading conclusions and inaccurate statements keep reappearing in literature; see [13, Remark 2.1], [16, Theorem 2.5], [7, Section 9], [9, p. 18 and Exercise 2.8.5] and [10, p. 261].

2 The CSI convergence bound based on scaled and shifted Chebyshev polynomial

The importance of Chebyshev polynomials in numerical computations was pointed out in the works of Flanders and Shortley [3], Lanczos [11] and Young [19]. This gave rise to the Chebyshev semi-iterative method (CSI) thoroughly described, e.g., in [18, Chapter 5], [20, Chapter 11] which was understood as an acceleration of the stationary Richardson iterations [14]. The k-th error of the CSI method can be written as

$$x - x_k = \frac{\chi_k(A)}{\chi_k(0)}(x - x_0) \tag{2}$$

where $\chi_k(\lambda)$ is the k-th shifted Chebyshev polynomial

$$\chi_k(\lambda) = \begin{cases} \cos\left(k \arccos\left(\dfrac{2\lambda - \lambda_N - \lambda_1}{\lambda_N - \lambda_1}\right)\right) & \text{for } \lambda \in [\lambda_1, \lambda_N], \\ \cosh\left(k \operatorname{arccosh}\left(\dfrac{2\lambda - \lambda_N - \lambda_1}{\lambda_N - \lambda_1}\right)\right) & \text{for } \lambda \notin [\lambda_1, \lambda_N] \end{cases} \tag{3}$$

and the method is optimal in a sense that $\chi_k(\lambda)/\chi_k(0)$ represents the unique solution of the minimization problem

$$\min_{\substack{\phi(0)=1 \\ \deg(\phi)\le k}} \max_{\lambda\in[\lambda_1,\lambda_N]} |\phi(\lambda)| \,. \tag{4}$$

Using the spectral decomposition of the HPD matrix $A = U\mathrm{diag}(\lambda_1, \ldots, \lambda_N)U^*$, $U^*U = UU^* = I$, where $0 < \lambda_1 < \ldots < \lambda_N$, $U = [u_1, \ldots, u_n]$, and using $|\chi_k(\lambda)| \leq 1$ for $\lambda \in [\lambda_1, \lambda_N]$, the relative A-norm of the error is in the CSI method bounded as

$$\frac{\|x - x_k\|_A}{\|x - x_0\|_A} \leq \left\| \frac{\chi_k(A)}{\chi_k(0)} \right\| = |\chi_k(0)|^{-1} \max_{j=1,\ldots,N} |\chi_k(\lambda_j)| \tag{5}$$

$$\leq |\chi_k(0)|^{-1} \max_{\lambda \in [\lambda_1, \lambda_N]} |\chi_k(\lambda)| = |\chi_k(0)|^{-1} \tag{6}$$

$$\leq 2 \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^k, \quad \kappa(A) = \frac{\lambda_N}{\lambda_1}, \quad k = 1, 2, \ldots, \tag{7}$$

where the last inequality is an easy consequence of the alternative definition of the Chebyshev polynomials (see, e.g., [15, Section 1.1]). The bound (7) for the CSI method was published explicitly in this form by Rutishauser [2, II.23] in 1959 and we see that it is based only on information about the extreme eigenvalues λ_1 and λ_N. It should be emphasized that Rutishauser then trivially concluded that since the CG method minimizes the A-norm of the error, the bound (7) is valid also for the CG method.

Using the spectral decomposition of A, we can for the CG approximations write

$$\|x - x_k\|_A = \min_{\substack{\varphi(0)=1 \\ \deg(\varphi) \leq k}} \|\varphi(A)(x - x_0)\|_A = \min_{\substack{\varphi(0)=1 \\ \deg(\varphi) \leq k}} \left\{ \sum_{j=1}^{N} |\xi_j|^2 \lambda_j \varphi^2(\lambda_j) \right\}^{1/2} \tag{8}$$

$$\leq \|x - x_0\|_A \min_{\substack{\varphi(0)=1 \\ \deg(\varphi) \leq k}} \max_{j=1,\ldots,N} |\varphi(\lambda_j)|, \tag{9}$$

where $|\xi_j|$ represents the size of the component of the initial error $x - x_0$ in the direction of the eigenvector u_j corresponding to λ_j, i.e., $x - x_0 = \sum_{j=1}^{N} \xi_j u_j$. The formula (8) shows that the error of CG computations is based on information about *all* eigenvalues of A and *all* projections of the initial error on the corresponding invariant subspaces. Naturally

$$\min_{\substack{\varphi(0)=1 \\ \deg(\varphi) \leq k}} \max_{j=1,\ldots,N} |\varphi(\lambda_j)| \leq \min_{\substack{\phi(0)=1 \\ \deg(\phi) \leq k}} \max_{\lambda \in [\lambda_1, \lambda_N]} |\phi(\lambda)| = |\chi_k(0)|^{-1}, \tag{10}$$

where the right part depends only on the extreme eigenvalues of A.

3 Composite polynomial bounds and their relevance in finite precision computations

In order to describe the superlinear convergence, Axelsson [1] and Jennings [8] consider in the presence of m outlying large eigenvalues the following polynomial

$$q_m(\lambda) = \prod_{j=N-m+1}^{N} \left(1 - \frac{\lambda}{\lambda_j} \right). \tag{11}$$

Using $|q_m(\lambda_j)| \leq 1$ for $j = 1, \ldots, N - m$ and the composite polynomial

$$q_m(\lambda)\chi_{k-m}(\lambda)/\chi_{k-m}(0), \tag{12}$$

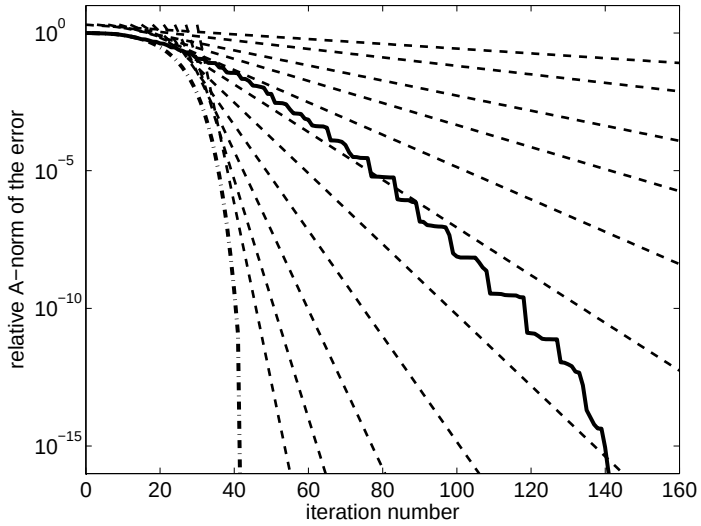


Figure 1: The sequence of the composite polynomial bounds (13) (dashed lines) for increasing number of large outlying eigenvalues ($m = 0, 3, 6, \ldots$) is compared with the results of finite precision CG computations (bold solid line) and exact CG behaviour (dash-dotted line).

where $\chi_{k-m}(\lambda)$ denotes the Chebyshev polynomial of degree $k-m$ shifted to the interval $[\lambda_1, \lambda_{N-m}]$ results in the bound

$$\frac{\|x - x_k\|_A}{\|x - x_0\|_A} \leq 2\left(\frac{\sqrt{\kappa_m(A)} - 1}{\sqrt{\kappa_m(A)} + 1}\right)^{k-m}, \quad k = m, m+1, \ldots, \tag{13}$$

where $\kappa_m(A) \equiv \lambda_{N-m}/\lambda_1$ is the so-called *effective condition number*. This quantity is typically substantially smaller than the condition number $\kappa(A)$ which indicates a possibly faster convergence after m initial iterations (cf. [12, Theorem 5.6.9]).

This is true, however, *only in exact arithmetic*. *In finite precision computations* this bound must, in general, fail. Motivating example is presented in Figure 1. While the bounds with the increasing number of the largest eigenvalues considered as outliers form close envelope of the exact CG behaviour (dash-dotted line), *none* of the straight lines describes the finite precision behaviour (bold line). The failure of the composite polynomial bound (13) in finite precision CG computations can occur even for a small size and/or conditioning of the problem. The explanation of the point is based on the backward-like analysis done by Greenbaum [5]. For more details we refer to [4] and [12, Chapter 5].

Acknowledgement: This work has been supported by the ERC-CZ project LL1202 and the GAUK grant 695612.

References

[1] O. Axelsson: *A class of iterative methods for finite element equations*. Comput. Methods Appl. Mech. Engrg. 9 (2), 1976, pp. 123–127.

[2] M. Engeli, T. Ginsburg, H. Rutishauser, E. Stiefel: *Refined iterative methods for computation of the solution and the eigenvalues of self-adjoint boundary value problems*. Mitt. Inst. Angew. Math. Zürich. 8, 1959.

[3] D. A. Flanders, G. Shortley: *Numerical determination of fundamental modes*. J. Appl. Phys. 21, 1950, pp. 1326–1332.

[4] T. Gergelits, Z. Strakoš: *Composite convergence bounds based on Chebyshev polynomials and finite precision conjugate gradient computations*. Submitted to Numerical Algorithms, December, 2012.

[5] A. Greenbaum: *Behaviour of slightly perturbed Lanczos and conjugate-gradient recurrences*. Linear Algebra Appl. 113, 1989, pp. 7–63.

[6] M. R. Hestenes, E. Stiefel: *Methods of conjugate gradients for solving linear systems*. J. Research Nat. Bur. Standards 49, 1952, pp. 409–436.

[7] I. C. F. Ipsen, C. D. Meyer: *The idea behind Krylov methods.* Amer. Math. Monthly 105 (10), 1998, pp. 889–899.

[8] A. Jennings: *Influence of the eigenvalue spectrum on the convergence rate of the conjugate gradient method*. J. Inst. Math. Appl. 20 (1), 1977, pp. 61–72.

[9] C. T. Kelley: *Iterative methods for linear and nonlinear equations.* Vol. 16 of Frontiers in Applied Mathematics, SIAM, 1995.

[10] D. Kratzer, S. V. Parter, M. Steuerwalt: *Block splittings for the conjugate gradient method*. Comput. & Fluids 11 (4), 1983, pp. 255–279.

[11] C. Lanczos: *Chebyshev polynomials in the solution of large-scale linear systems*. In: Proceedings of the Association for Computing Machinery, Toronto, 1952, Sauls Lithograph Co. (for the Association for Computing Machinery), Washington, D. C., 1953, pp. 124–133.

[12] J. Liesen, Z. Strakoš: *Krylov subspace methods: principles and analysis*. Numerical Mathematics and Scientific Computation, Oxford University Press, 2012.

[13] K.-A. Mardal, R. Winther: *Preconditioning discretizations of systems of partial differential equations*. Numer. Linear Algebra Appl. 18 (1), 2011, pp. 1–40.

[14] L. F. Richardson: *The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam*. Phil. Trans. Roy. Soc. London, A, 210, 1911, pp. 307–257.

[15] T. J. Rivlin: *The Chebyshev polynomials*. Pure and Applied Mathematics, Wiley-Interscience [John Wiley & Sons], 1974.

[16] D. A. Spielman, J. Woo: *A note on preconditioning by low-stretch spanning trees*. Computing Research Repository, 2009.

[17] A. van der Sluis, H. A. van der Vorst: *The rate of convergence of conjugate gradients*. Numer. Math. 48 (5), 1986, pp. 543–560.

[18] R. S. Varga: *Matrix iterative analysis*. Expanded ed., vol. 27 of Springer Series in Computational Mathematics, Springer-Verlag, 2000 (Originally published 1962).

[19] D. M. Young: *On Richardson's method for solving linear systems with positive definite matrices*. J. Math. Physics 32, 1954, pp. 243–255.

[20] D. M. Young: *Iterative solution of large linear systems*. Academic Press, New York, 1971.

FLLOP: a massively parallel QP solver

V. Hapla, D. Horák, F. Staněk

IT4Innovations & DAM, VŠB-Technical University of Ostrava

1 Quadratic programming

Discretization of most engineering problems, describable as partial differential equations (PDE), leads to large sparse linear systems of equations (perhaps using some linearization technique). However, problems that can be expressed as elliptic variational inequalities, such as those describing the equilibrium of elastic bodies in mutual contact, are more naturally discretized to quadratic programming problems (quadratic programs, QP). They take this canonical form:

$$\min_{\mathbf{x}} \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} - \mathbf{b}^T\mathbf{x} \tag{1}$$

$$\text{subject to } \mathbf{B}_E\mathbf{x} = \mathbf{c}_E, \tag{2}$$

$$\mathbf{B}_I\mathbf{x} \leq \mathbf{c}_I. \tag{3}$$

We will consider problems with a symmetric positive definite *Hessian* $\mathbf{A}$. The vector $\mathbf{b}$ is called *righ-hand side*. QP can be thought of as a generalization of a linear system of equations with prescribed *equality* (2) and *inequality* (3) *constraints*. Very common special case of inequality constraints are *box constraints*

$$\mathbf{l} \leq \mathbf{x} \leq \mathbf{u} \tag{4}$$

where elements of $\mathbf{l}$ have values from $\mathbb{R} \cup \{-\infty\}$ and elements of $\mathbf{u}$ have values from $\mathbb{R} \cup \{+\infty\}$. Note that *unconstrained* QP ($\mathbf{B}_E, \mathbf{c}_E, \mathbf{B}_I, \mathbf{c}_I$ are zero objects) has the same solution as a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$.

2 FLLOP design

We present here our novel software package for solution of QP called FLLOP (FETI Light Layer On top of PETSc). It is an extension of PETSc framework. PETSc (Portable, Extensible Toolkit for Scientific Computation) [8] is a suite of data structures and routines for the parallel solution of scientific applications modelled by PDE.

FLLOP is carefully designed to be user-friendly while remaining efficient and targeted to HPC. The typical workflow looks like this:

1. natural specification of the QP by the user,
2. a user-specified series of QP transformations,
3. automatic or manual choice of a sensible solver,
4. solution of the most derived QPs by the chosen solver,
5. a series of backward transformations to get a solution of the original QP (triggered by the solver).

Specification of the QP. A class used to specify a QP problem is called simply `QP`. It is a data structure containing at least the Hessian matrix $\mathbf{A}$, right hand side $\mathbf{b}$ and the solution vector $\mathbf{x}$ (these objects are called `Operator, Rhs, SolutionVector` in FLLOP). Additionaly, any combination of these constraints can be specified:

1. equality constraints (`Beq`, `ceq`),
2. inequality constraints (`Bineq`, `cineq`),
3. box constraints (`lb`, `ub`).

Objects that are not specified (i.e. set to `PETSC_NULL`) are handled as zero objects.

QP transformations and backward transformations. A QP transformation derives a new QP from the given QP. They allow use of efficient solvers but are themselves solver-neutral. Currently, we have these in FLLOP:

1. dualization (`Dualize`),
2. homogenization of the equality constraints (`HomogenizeEq`),
3. enforce $\mathbf{B}_E\mathbf{x} = \mathbf{o}$ using penalty or projector onto the kernel (`EnforceEq`).

For instance, *homogenization of the equality constraints* transforms a QP with general equality constraints $\mathbf{B}_E\mathbf{x} = \mathbf{c}_E$ to a new one with homogeneous equality constraints $\mathbf{B}_E\mathbf{x} = \mathbf{o}$. It consinsts in finding a particular solution $\widetilde{\mathbf{x}}$ that satisfies $\mathbf{B}_E\widetilde{\mathbf{x}} = \mathbf{c}_E$. The right hand side $\mathbf{b}_0$ and the box constraints $(\mathbf{l}_0, \mathbf{u}_0)$ of the original problem are then transformed to $\mathbf{b}_1 = \mathbf{b} - \mathbf{A}\widetilde{\mathbf{x}}$ and $(\mathbf{l}_1, \mathbf{u}_1) = (\mathbf{l}_0 - \widetilde{\mathbf{x}}, \mathbf{u}_0 - \widetilde{\mathbf{x}})$, respectively.

In FLLOP, every QP transformation creates a new instance `QP1` of the `QP` class based on the original `QP0`. The data are either (1) shared between `QP0` and `QP1`, (2) copied from `QP0`, modified and stored to `QP1`. Furthermore, links between `QP0` and `QP1` are created: `QP0` has a *child* link to `QP1`, `QP1` has a *parent* link to `QP0`. Thus, sort of doubly linked list is generated where every node is a QP.

Of course, the solution $\mathbf{x}_1$ of the new QP is not equal to a solution $\mathbf{x}_0$ of the original one – we have to carry out a proper*backward transformation* of the solution. In the above-mentioned case, it holds that $\mathbf{x}_0 = \mathbf{x}_1 + \widetilde{\mathbf{x}}$. In FLLOP, we use a notion of *post-solve function* for this purpose. It is a pointer to function that computes the solution of the parent QP based on solution of the children QP; it is injected to the child QP by the transformation function. The post-solve functions connected to a given series of QP transformations are called by the solver in the reversed order of those to get the solution of the very original problem.

We also need to store somewhere the auxiliary data created by the transformation and needed by the backward transformation ($\widetilde{\mathbf{x}}$ in our case). For this purpose so called *post-solve context* is used; it is a `void` pointer, also injected to a child QP. Note that the child QP does not use nor know anything about the post-solve function and context; they are only set by the transformation function and accessed by the solver.

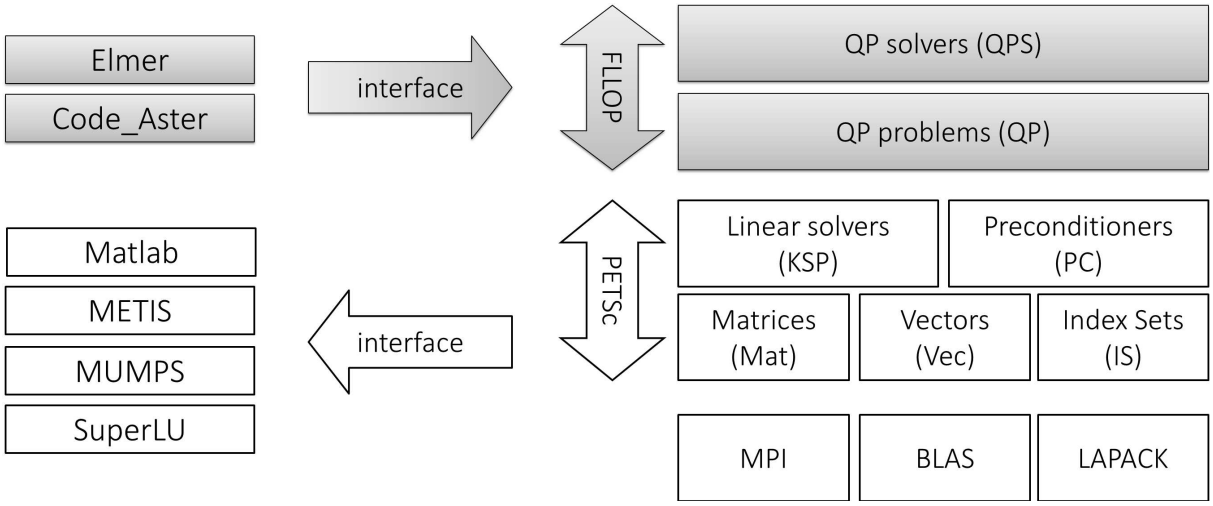


Figure 1: Hierarchy of PETSc and FLLOP and their relations to external software.

3 FETI in FLLOP

FLLOP was originally developed as an implementation of the FETI domain decomposition method. However, most recent advances in the design of FLLOP allow more general use. There are essentially two levels of generalization:

1. It allows to apply FETI to variational inequalities (e.g. contact problems).
2. The algebraic part of FETI computation is generalized to a specific combination of data structures, QP transformations, direct and iterative solvers. However, these ingredients can make sense also out of the original FETI method. For instance, dualization can be useful also for undecomposed problems; on the other hand, decomposed problems can be solved without dualization.

Acknowledgements

This publication was supported by the 'Projects of major infrastructures for research, development and innovation' of Ministry of Education, Youth and Sports (LM2011033), the 'IT4Innovations Centre of Excellence' project (CZ.1.05/1.1.00/02.0070) funded by the European Regional Development Fund, by the project 'SPOMECH – Creating a multidisciplinary R&D team for reliable solution of mechanical problems' (CZ.1.07/2.3.00/20.0070) within Operational Programme 'Education for competitiveness' funded by Structural Funds of the European Union and state budget of the Czech Republic and by the project 'Cooperation for Future' (CZ.1.07/2.4.00/31.0035).

The research has also been supported by the grants: HPC-Europa2 project funded by the European Commission - DG Research in the Seventh Framework Programme under grant agreement No. 228398, PRACE 2IP project receiving funding from the EU's Seventh Framework Programme (FP7/2007-2013) under grant agreement No. RI-283493, and by the Ministry of Education, Youth and Sports under grant agreement No. MSM 6198910027.

References

[1] Z. Dostál: *Optimal quadratic programming algorithms, with applications to variational inequalities*. 1st edition. SOIA 23. Springer US, New York, 2009.

[2] V. Hapla, D. Horak, M. Merta: *Use of direct solvers in TFETI massively parallel implementation*. In: Proceedings of PARA 2012: Workshop on State-of-the-Art in Scientific and Parallel Computing, accepted 2012.

[3] V. Hapla, D. Horak: *TFETI coarse space projectors parallelization strategies*. In: Proceedings of PPAM 2011, Lecture Notes in Computer Science, 7203 LNCS (Part 1), pp. 152–162, ISSN: 03029743, ISBN: 978-364231463-6, DOI: 10.1007/978-3-642-31464-3_16, 2012.

[4] Z. Dostál, D. Horák, R. Kučera: *Total FETI – an easier implementable variant of the FETI method for numerical solution of elliptic PDE*. Commun. in Numerical Methods in Engineering 22, 2006, pp. 1155–1162.

[5] T. Kozubek, V. Vondrák, M. Menšík, D. Horák, Z. Dostál, V. Hapla, P. Kabelíková, M. Čermák: *Total FETI domain decomposition method and its massively parallel implementation*. Accepted in *Advances in Engineering Software*, 2011.

[6] S. Balay, J. Brown, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, H. Zhang: *PETSc users manual*. Tech. Rep. ANL-95/11 – Revision 3.2, Argonne National Laboratory, 2011.

[7] S. Balay, W. D. Gropp, L. C. McInnes, B. F. Smith: *Efficient management of parallelism in object oriented numerical software libraries*. In: Modern Software Tools in Scientific Computing, E. Arge, A. M. Bruaset, H. P. Langtangen, (Eds), Birkhäuser Press, 1997, pp. 163–202.

[8] PETSc Web page, `http://www.mcs.anl.gov/petsc/`

[9] FLLOP Web Page, `http://spomech.vsb.cz/feti/`

On a pathfollowing method for solving the contact problem with Coulomb friction

J. Haslinger, V. Janovský, R. Kučera

[1,2] Faculty of Mathematics and Physics, Charles University, Prague
[3] Department of Mathematics and Descriptive Geometry, VŠB-TU, Ostrava

1 Discrete static contact problems with Coulomb friction

Consider deformable bodies in mutual contact. The relevant mathematical description consists in modelling both non-penetration conditions and a friction law. The widely accepted Coulomb friction law represents a serious mathematical and numerical problem.

In particular, we consider the static contact problem with Coulomb friction on a planar domain. The problem is uniquely solvable, provided that the friction coefficient $\mathcal{F} > 0$ is sufficiently small, see e.g. [2]. Note that no essential contribution was made concerning solvability of this problem for general data. Nevertheless, engineers had always solved this important problem numerically, regardless unresolved theoretical issues. In the natural finite element (FEM) approximation, the discrete problem has always a solution, disregarding the size of $\mathcal{F}$, see [6, 4, 9].

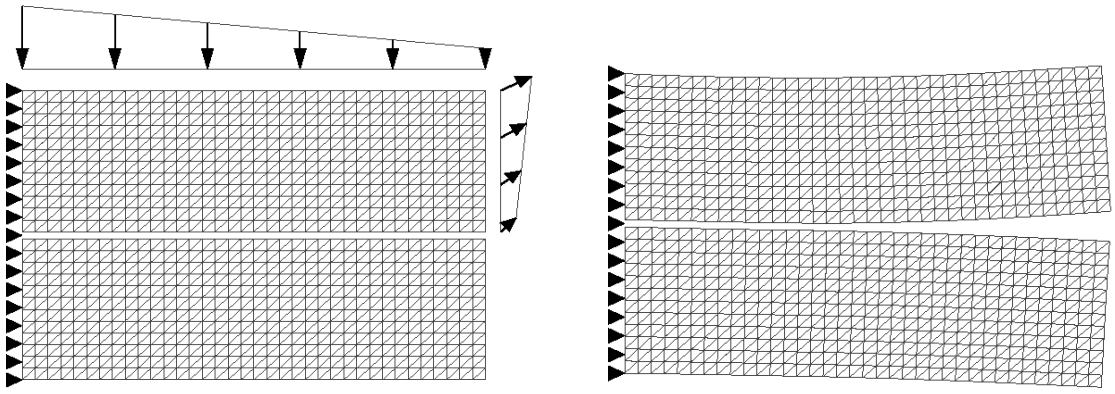


Figure 1: Contact of two elastic bodies Ω^1 (the upper body) and Ω^2, along the contact boundary Γ_c. The loading is due to the surface traction. On the right: Resulting displacements.

We consider a particular geometry, see Figure 1. The FEM approximation (linear elements) yields the following *primal-dual* discrete state problem:

$$\boldsymbol{K}\boldsymbol{u} + \boldsymbol{N}^\top \boldsymbol{\lambda}_\nu + \boldsymbol{T}^\top \boldsymbol{\lambda}_t = \boldsymbol{f}, \tag{1}$$

$$\boldsymbol{N}\boldsymbol{u} \le 0,\ \boldsymbol{\lambda}_\nu \ge 0,\ \boldsymbol{\lambda}_\nu^\top \boldsymbol{N}\boldsymbol{u} = 0, \tag{2}$$

$$\left.\begin{array}{l} |\lambda_{t,i}| \le \mathcal{F}\lambda_{n,i}, \\ |\lambda_{t,i}| < \mathcal{F}\lambda_{n,i} \Rightarrow (\boldsymbol{T}\boldsymbol{u})_i = \boldsymbol{0}, \\ |\lambda_{t,i}| = \mathcal{F}\lambda_{n,i} \Rightarrow \exists\, c_{t,i} \ge 0 :\ (\boldsymbol{T}\boldsymbol{u})_i = c_{t,i}\lambda_{t,i}, \end{array}\right\} \quad i = 1, \dots, m, \tag{3}$$

where $(\boldsymbol{u}, \boldsymbol{\lambda}_\nu, \boldsymbol{\lambda}_t) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^m$. Here $\boldsymbol{u}$ approximates displacement field, n is dofs. $\boldsymbol{\lambda}_\nu$ and $\boldsymbol{\lambda}_t$ approximate normal and tangential stress components along the contact boundary Γ_c, m is

the number of contact nodes. Data of the model: $\boldsymbol{K} \in \mathbb{R}^{n\times n}$ is positive definite stiffness matrix, $\boldsymbol{N}, \boldsymbol{T} \in \mathbb{R}^{m\times n}$ are full rank matrices (the actions of distributed contact forces along normal and tangential directions), $\boldsymbol{f} \in \mathbb{R}^n$ are nodal forces.

The inequalities (2) and (3) can be equivalently written as

$$\boldsymbol{\lambda}_\nu - P_{\mathbb{R}^m_+}(\boldsymbol{\lambda}_\nu + \rho \boldsymbol{N}\boldsymbol{u}) = \boldsymbol{0} \quad \text{and} \quad \boldsymbol{\lambda}_t - P_{[-\mathcal{F}\boldsymbol{\lambda}_\nu, \mathcal{F}\boldsymbol{\lambda}_\nu]}(\boldsymbol{\lambda}_t + \rho \boldsymbol{T}\boldsymbol{u}) = \boldsymbol{0},$$

respectively, where $P_{\mathbb{R}^m_+}$ and $P_{[-\mathcal{F}\boldsymbol{\lambda}_\nu, \mathcal{F}\boldsymbol{\lambda}_\nu]}$ are suitable projectors, see [3]. Parameter $\rho > 0$ is arbitrary, but fixed (e.g., $\rho = 1$). Therefore, solving (1)–(3) is equivalent to finding roots $\boldsymbol{y} = (\boldsymbol{u}, \boldsymbol{\lambda}_\nu, \boldsymbol{\lambda}_t) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^m$ of the equation

$$G(\boldsymbol{y}) \equiv \begin{pmatrix} \boldsymbol{K}\boldsymbol{u} + \boldsymbol{N}^\top \boldsymbol{\lambda}_\nu + \boldsymbol{T}^\top \boldsymbol{\lambda}_t \\ \boldsymbol{\lambda}_\nu - P_{\mathbb{R}^m_+}(\boldsymbol{\lambda}_\nu + \rho \boldsymbol{N}\boldsymbol{u}) \\ \boldsymbol{\lambda}_t - P_{[-\mathcal{F}\boldsymbol{\lambda}_\nu, \mathcal{F}\boldsymbol{\lambda}_\nu]}(\boldsymbol{\lambda}_t + \rho \boldsymbol{T}\boldsymbol{u}) \end{pmatrix} = \begin{pmatrix} \boldsymbol{f} \\ 0 \\ 0 \end{pmatrix}, \tag{4}$$

where $\boldsymbol{y} = (\boldsymbol{u}, \boldsymbol{\lambda}_\nu, \boldsymbol{\lambda}_t) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^m$. The mapping $G : \mathbb{R}^{n+2m} \mapsto \mathbb{R}^{n+2m}$ is continuous and piecewise smooth. In particular, it is *piecewise affine*, see e.g. [10] for the notion.

2 The semi-smooth Newton method

For solving (4), we apply the Newton iterations. Due to nature of the operator G, semi-smooth methods are applicable, see e.g. [7]. Let $\mathcal{M} = \{1, 2, \ldots, m\}$ be the set of all indices of contact points: Given $\boldsymbol{y} = (\boldsymbol{u}, \boldsymbol{\lambda}_\nu, \boldsymbol{\lambda}_t) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^m$, we define the *inactive* sets $\mathcal{I}_\nu = \mathcal{I}_\nu(\boldsymbol{y})$, $\mathcal{I}_t^+ = \mathcal{I}_t^+(\boldsymbol{y})$, $\mathcal{I}_t^- = \mathcal{I}_t^-(\boldsymbol{y})$ by

$$\begin{aligned} \mathcal{I}_\nu &= \{i \in \mathcal{M} : \lambda_{\nu,i} + \rho(\mathbf{N}\mathbf{u})_i < 0\}, \\ \mathcal{I}_t^+ &= \{i \in \mathcal{M} : \lambda_{t,i} + \rho(\mathbf{T}\mathbf{u})_i - \mathcal{F}\lambda_{\nu,i} > 0\}, \\ \mathcal{I}_t^- &= \{i \in \mathcal{M} : \lambda_{t,i} + \rho(\mathbf{T}\mathbf{u})_i + \mathcal{F}\lambda_{\nu,i} > 0\}, \end{aligned}$$

and the *active* sets $\mathcal{A}_\nu = \mathcal{A}_\nu(\boldsymbol{y})$, $\mathcal{A}_t = \mathcal{A}_t(\boldsymbol{y})$ as their complements:

$$\mathcal{A}_\nu = \mathcal{M} \setminus \mathcal{I}_\nu, \quad \mathcal{A}_t = \mathcal{M} \setminus (\mathcal{I}_t^+ \cup \mathcal{I}_t^-).$$

Let us introduce the indicator matrix $\boldsymbol{D}_\mathcal{S} \in \mathbb{R}^{m\times m}$ of $\mathcal{S} \subset \mathcal{M}$ as follows:

$$\boldsymbol{D}_\mathcal{S} = diag(s_1, \ldots, s_m), \quad s_i = \begin{cases} 1, & i \in \mathcal{S}, \\ 0, & i \in \mathcal{M} \setminus \mathcal{S}. \end{cases}$$

We observe that

$$G(\boldsymbol{y}) = \begin{pmatrix} \boldsymbol{K}\boldsymbol{u} + \boldsymbol{N}^\top \boldsymbol{\lambda}_\nu + \boldsymbol{T}^\top \boldsymbol{\lambda}_t \\ \boldsymbol{\lambda}_\nu - \boldsymbol{D}_{\mathcal{A}_\nu}(\boldsymbol{\lambda}_\nu + \rho \boldsymbol{N}\boldsymbol{u}) \\ \boldsymbol{\lambda}_t - \boldsymbol{D}_{\mathcal{A}_t}(\boldsymbol{\lambda}_t + \rho \boldsymbol{T}\boldsymbol{u}) - \boldsymbol{D}_{\mathcal{I}_t^+}\mathcal{F}\boldsymbol{\lambda}_\nu + \boldsymbol{D}_{\mathcal{I}_t^-}\mathcal{F}\boldsymbol{\lambda}_\nu \end{pmatrix} = J(\boldsymbol{y})\,\boldsymbol{y}\,,$$

where

$$J(\boldsymbol{y}) \equiv \left(\begin{array}{c|c|c} \boldsymbol{K} & \boldsymbol{N}^\top & \boldsymbol{T}^\top \\ \hline -\rho \boldsymbol{D}_{\mathcal{A}_\nu}\boldsymbol{N} & \boldsymbol{D}_{\mathcal{I}_\nu} & \boldsymbol{0} \\ \hline -\rho \boldsymbol{D}_{\mathcal{A}_t}\boldsymbol{T} & \mathcal{F}(\boldsymbol{D}_{\mathcal{I}_t^-} - \boldsymbol{D}_{\mathcal{I}_t^+}) & \boldsymbol{D}_{\mathcal{I}_t^+ \cup \mathcal{I}_t^-} \end{array}\right). \tag{5}$$

Algorithm *SSNM:* Denote $\boldsymbol{F} \in \mathbb{R}^{n+2m}$, $\boldsymbol{F} \equiv (\boldsymbol{f}, 0, 0) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^m$, the right-hand side of (4). Set the tolerance $\mathcal{F}repsilon > 0$. Let $\boldsymbol{y}^{(0)} \in \mathbb{R}^{n+2m}$, $\rho > 0$, $k := 1$.

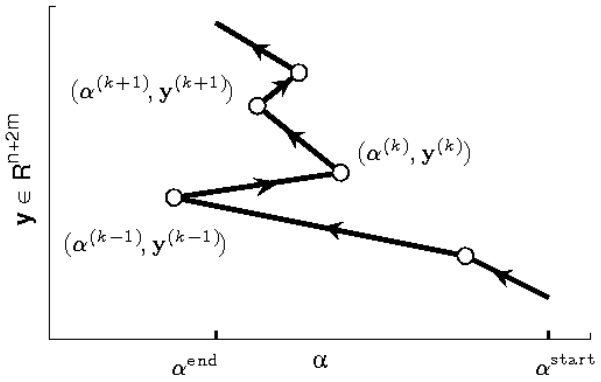


Figure 2: Solution path. For a fixed α, we may encounter up to five intersection points on the path. They are related to five different solutions of equation (4) for the same right-hand side.

(i) Define the inactive/active sets related to $\boldsymbol{y}^{(k-1)}$. Assembly the relevant $J(\boldsymbol{y}^{(k-1)})$.

(ii) Compute $\boldsymbol{y}^{(k)}$ by solving the linear system $J(\boldsymbol{y}^{(k-1)})\,\boldsymbol{y}^{(k)} = \boldsymbol{F}$.

(iii) If $||\boldsymbol{y}^{(k)} - \boldsymbol{y}^{(k-1)}||/||\boldsymbol{y}^{(k)}|| \leq \mathcal{F}repsilon$, return $\boldsymbol{y} := \boldsymbol{y}^{(k)}$.

(iv) Set $k := k + 1$ and go to step *(i)*.

In the case of convergence, let $\boldsymbol{y} = SSNM(\boldsymbol{y}^{(0)}, \boldsymbol{f})$ as a numerical solution of problem (4).

3 Continuation

Consider the Coulomb friction model (1)-(3), i.e. (4), assuming that $\boldsymbol{f} = \boldsymbol{f}(\alpha)$ depends on a scalar parameter α. We impose a continuous loading regime and seek for a *continuous* response of the model. In particular, we consider a linear *loading path*

$$\boldsymbol{f}(\alpha) = (1 - \alpha)\boldsymbol{f}_1 + \alpha\boldsymbol{f}_2, \quad \alpha \in \mathbb{R},$$

where $\boldsymbol{f}_1 \in \mathbb{R}^n$ and $\boldsymbol{f}_2 \in \mathbb{R}^n$ are given. The resulting *solution path* is a curve in $\mathbb{R} \times \mathbb{R}^{n+2m}$, see a qualitative sketch in Figure 2. It consists of *oriented linear branches*, connected by *transition points*.

- In order to follow the oriented linear branches, we implemented *tangent continuation*, see [1], Algorithm 4.25, with *SSNM* as a corrector. We implemented an adaptive step-size control.
- In order to detect transition points, we introduced *branching* and *orientation* indicators. The idea is to modify inactive sets $\mathcal{I}$ properly.

Details will be given in [5]. The actual computations are illustrated in Figure 3.

Acknowledgement: This work was supported by the Grant Agency of the Czech Republic (grant No. P201/12/0671).

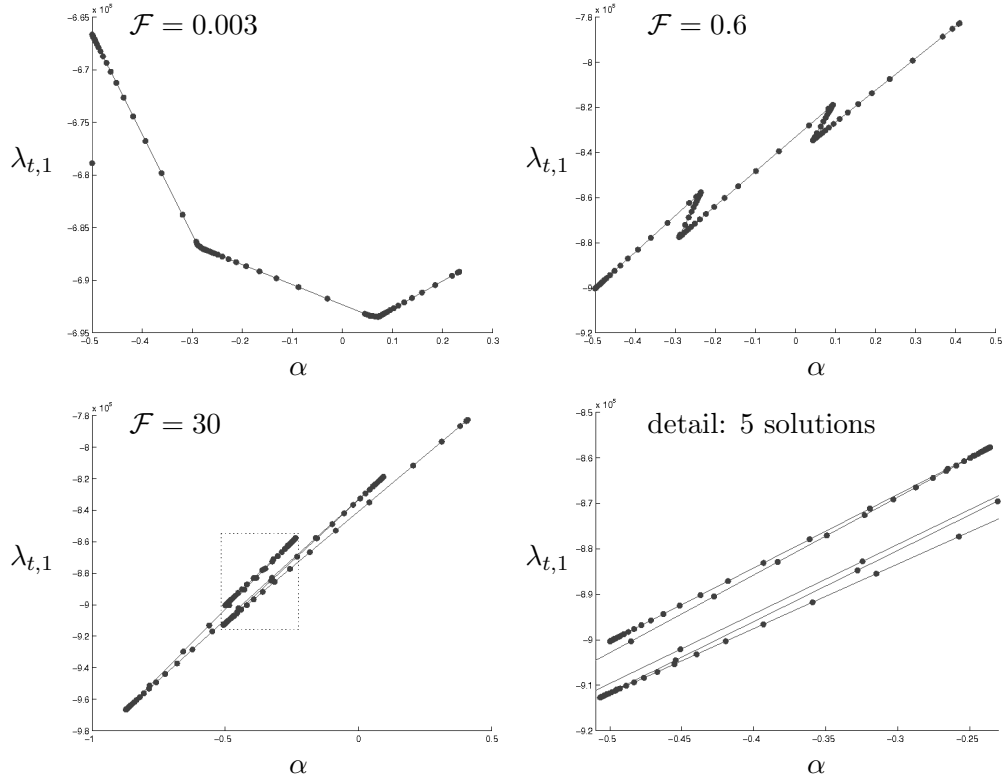


Figure 3: Discretization: $n = 1320$, $m = 30$. Plots: Parameter α vs. the solution component $\lambda_{t,1}$, for selected friction coefficients $\mathcal{F}$.

References

[1] P. Deuflhart, A. Hohmann: *Numerical analysis in modern scientific computing.* Texts in Applied Mathematics, Springer Verlag, New York, 2003.

[2] C. Eck, J. Jarušek: *Existence results for the static contact problems with Coulomb friction.* Math. Models Methods Appl. Sci. 8, 1997, pp. 445–468.

[3] F. Facchinei, J. Pang: *Finite-dimensional variational inequalities and complementarity problems.* Springer Series in Operations Research xxxiii, New York, 2003.

[4] J. Haslinger, V. Janovský , T. Ligurský : *Qualitative analysis of solutions to discrete static contact problems with Coulomb friction.* Comp. Meth. Appl. Mech. Engrg. 205–208, 2012, pp. 149–161.

[5] J. Haslinger, V. Janovský , R. Kučera : *Path-following the static contact problem with Coulomb friction.* Submitted to Appl. Math.

[6] P. Hild, Y. Renard: *Local uniqueness and continuation of solutions for the discrete Coulomb friction problem in elastostatics.* Quart. Appl. Math. 63, 2005, pp. 553–573.

[7] K. Ito, K. Kunisch: *Semi-smooth Newton methods for the Signorini problem.* Appl. Math. 53, 2009, pp. 455–468.

[8] V. Janovský , T. Ligurský : *Computing non unique solutions of the Coulomb friction problem.* Math. Comput. Simul. 82, 2012, pp. 2047–2061.

[9] T. Ligurský : *Theoretical analysis of discrete contact problems with Coulomb friction.* Appl. Math. 57, 2012, pp. 263–295.

[10] Scholtes, S.: *Introduction to piecewise differentiable equations.* SpringerBriefs in Optimalization, Springer, Berlin, 2012.

Shape sensitivity analysis in discretized 2D contact problems with Coulomb friction and a solution-dependent coefficient of friction

J. Haslinger, J. V. Outrata, R. Pathó

[1,3] Department of Numerical Mathematics, Charles University in Prague
[2] Institute of Information Theory and Automation AS CR, Prague

1 Introduction

The contribution deals with shape optimization of an elastic body that is unilaterally supported by a rigid foundation. We aim at extending existing results [1, 2] to the more general case when the coefficient of friction $\mathcal{F}$ may depend on solution, namely on the magnitude of the unknown tangential displacement: $\mathcal{F} = \mathcal{F}(|\boldsymbol{u}_\tau|)$. As state problem we consider the two-dimensional, discretized Signorini problem with Coulomb friction, but in contrast to [1], the coefficient of friction is a function of the unknown solution. In particular, we concentrate on deriving first order sensitivities of the displacement field and normal contact stresses along the contact boundary. This will be done in the fashion of [2] and [4], namely, using the generalized differential calculus of Mordukhovich ([5]).

2 The state problem

Let $a, b > 0$ be given and let an elastic body be represented by the domain $\Omega(\alpha) := \{(x_1, x_2) \in \mathbb{R}^2 \mid 0 < x_1 < a,\ \alpha(x_1) < x_2 < b\}$, where

$$\alpha \in \mathcal{U}_{ad} := \{\alpha \in C^{0,1}([0,a]) \mid 0 \le \alpha \le C_0,\ ||\alpha'||_{L^\infty(0,a)} \le C_1,\ C_2 \le \text{meas}\ \Omega(\alpha) \le C_3\}. \quad (1)$$

It is implicitly assumed in (1) that $\mathcal{U}_{ad} \neq \emptyset$. Let $\partial\Omega(\alpha)$ be split into three non-empty, disjoint parts Γ_u, Γ_P and $\Gamma_c(\alpha)$ with different boundary conditions: on Γ_u the body is fixed, while surface tractions of density $\boldsymbol{P} = (P_1, P_2)$ act along Γ_P. On $\Gamma_c(\alpha) = \text{Gr}\,\alpha$, representing the contact part of $\partial\Omega(\alpha)$, the body is *unilaterally supported* by the perfectly rigid foundation $\Xi := \{(x_1, x_2) \mid x_2 \le 0\}$. In addition to the non-penetration conditions, we shall consider effects of friction between $\Omega(\alpha)$ and Ξ. We use the local *Coulomb friction law*, but with a coefficient of friction $\mathcal{F}$ which *depends* on the solution:

$$\left.\begin{array}{lll} u_1 = 0 & \Longrightarrow & |T_1(\boldsymbol{u})| \le \mathcal{F}(0) T_2(\boldsymbol{u}) \\ u_1 \neq 0 & \Longrightarrow & T_1(\boldsymbol{u}) = -\text{sgn}(u_1)\mathcal{F}(|u_1|) T_2(\boldsymbol{u}) \end{array}\right\} \text{ on } \Gamma_c(\alpha),$$

where $\boldsymbol{T}(\boldsymbol{u}) = (T_1(\boldsymbol{u}), T_2(\boldsymbol{u})) : \partial\Omega(\alpha) \to \mathbb{R}^2$ stands for the stress vector associated with the displacement field $\boldsymbol{u} = (u_1, u_2) : \Omega(\alpha) \to \mathbb{R}^2$. The equilibrium state of $\Omega(\alpha)$ is characterized by a displacement $\boldsymbol{u}$ that satisfies the system of linear equilibrium equations in $\Omega(\alpha)$, the classical boundary conditions on Γ_u, Γ_P and the unilateral and friction conditions on $\Gamma_c(\alpha)$.

Following the approximation procedure described in [1, 2, 4] one arrives at the following discretized Signorini problem with Coulomb friction and a solution-dependent coefficient of friction:

$$\left.\begin{aligned} &\text{For given } \boldsymbol{\alpha} \in U_{ad} \text{ find } (\boldsymbol{u}, \boldsymbol{\lambda}) \in \mathbb{R}^n \times \mathbb{R}^p_+ \text{ such that:} \\ &\langle \mathbb{A}(\boldsymbol{\alpha})\boldsymbol{u}, \boldsymbol{v} - \boldsymbol{u}\rangle_n + \sum_{i=1}^{p} \mathcal{F}(|(\boldsymbol{u}_\tau)_i|)\boldsymbol{\lambda}_i\big(|(\boldsymbol{v}_\tau)_i| - |(\boldsymbol{u}_\tau)_i|\big) \\ &\qquad \geq \langle \boldsymbol{L}(\boldsymbol{\alpha}), \boldsymbol{v} - \boldsymbol{u}\rangle_n + \langle \boldsymbol{\lambda}, \boldsymbol{v}_\nu - \boldsymbol{u}_\nu\rangle_p \quad \forall \boldsymbol{v} \in \mathbb{R}^n, \\ &\langle \boldsymbol{\mu} - \boldsymbol{\lambda}, \boldsymbol{u}_\nu + \boldsymbol{\alpha}\rangle_p \geq 0 \quad \forall \boldsymbol{\mu} \in \mathbb{R}^p_+, \end{aligned}\right\} \qquad (\mathcal{M}(\boldsymbol{\alpha}))$$

where $U_{ad} \subset \mathbb{R}^p_+$ is the *convex, compact* set of admissible design variables corresponding to the discretization of (1) (p denotes to the number of contact nodes) and $\boldsymbol{\lambda}$ is the Lagrange multiplier releasing the non-penetration constraint. It is related to the discretization of the normal contact stress $T_2(\boldsymbol{u})$. Further, $\boldsymbol{v}_\tau, \boldsymbol{v}_\nu \in \mathbb{R}^p$ stand for the subvectors of $\boldsymbol{v} \in \mathbb{R}^n$ consisting of the first and second components, respectively, of the displacement vector $\boldsymbol{v}$ at all contact nodes. As usual, $\mathbb{A}$ and $\boldsymbol{L}$ denote the stiffness matrix and load vector, respectively. Note, that $\mathbb{A} \in C^1(U_{ad}; \mathbb{R}^{n\times n})$ and $\boldsymbol{L} \in C^1(U_{ad}; \mathbb{R}^n)$.

In the rest of the paper we shall be dealing with the reduced form of $(\mathcal{M}(\boldsymbol{\alpha}))$ (cf. [1, 2]), which consists in eliminating all components of the displacement field $\boldsymbol{u}$ corresponding to the noncontact nodes of the finite element partition of the domain $\overline{\Omega}(\boldsymbol{\alpha})$. Thus one obtains a system of variational inequalities in terms of $\boldsymbol{u}_\tau$, $\boldsymbol{u}_\nu$, $\boldsymbol{\lambda}$ only:

$$\left.\begin{aligned} &\mathbf{0} \in \mathbb{A}_{\tau\tau}(\boldsymbol{\alpha})\boldsymbol{u}_\tau + \mathbb{A}_{\tau\nu}(\boldsymbol{\alpha})\boldsymbol{u}_\nu - \boldsymbol{L}_\tau(\boldsymbol{\alpha}) + Q_1(\boldsymbol{u}_\tau, \boldsymbol{\lambda}) \\ &\mathbf{0} = \mathbb{A}_{\nu\tau}(\boldsymbol{\alpha})\boldsymbol{u}_\tau + \mathbb{A}_{\nu\nu}(\boldsymbol{\alpha})\boldsymbol{u}_\nu - \boldsymbol{\lambda} - \boldsymbol{L}_\nu(\boldsymbol{\alpha}) \\ &\mathbf{0} \in \boldsymbol{u}_\nu + \boldsymbol{\alpha} + N_{\mathbb{R}^p_+}(\boldsymbol{\lambda}) \end{aligned}\right\} \qquad (2)$$

Introducing the state variable $\boldsymbol{y} = (\boldsymbol{u}_\tau, \boldsymbol{u}_\nu, \boldsymbol{\lambda}) \in \mathbb{R}^p \times \mathbb{R}^p_+ \times \mathbb{R}^p_+$, (2) can be written in the more compact form of one generalized equation (GE):

$$\mathbf{0} \in F(\boldsymbol{\alpha}, \boldsymbol{y}) + Q(\boldsymbol{y}), \qquad (3)$$

where F is continuously differentiable, $Q(\boldsymbol{y}) := \big(Q_1(\boldsymbol{y}_1, \boldsymbol{y}_3), \mathbf{0}, N_{\mathbb{R}^p_+}(\boldsymbol{y}_3)\big)^T$ and the multifunction Q_1 in (2) is defined as:

$$\big(Q_1(\boldsymbol{u}_\tau, \boldsymbol{\lambda})\big)_i := \mathcal{F}(|(\boldsymbol{u}_\tau)_i|)\boldsymbol{\lambda}_i \partial|(\boldsymbol{u}_\tau)_i| \quad \forall i = 1, \ldots, p.$$

Here "∂" stands for the convex subdifferential and $N_{\mathbb{R}^p_+}(\cdot)$ is the standard normal cone mapping in the sense of convex analysis.

Theorem 1. *Let $S : \boldsymbol{\alpha} \mapsto \{\boldsymbol{y} \mid \mathbf{0} \in F(\boldsymbol{\alpha}, \boldsymbol{y}) + Q(\boldsymbol{y})\}$ denote the control-to-state mapping and let $\mathcal{F} : \mathbb{R}_+ \to \mathbb{R}_+$ be bounded and Lipschitz continuous with sufficiently small upper bound and Lipschitz constant. Then S is single-valued and strongly regular in U_{ad}. Consequently, S is (locally) Lipschitz continuous.*

Proof. By modifying the proofs of Theorem 3.8, Proposition 3.11 and Theorem 3.13 in [2]. □

3 Shape optimization and sensitivity analysis

Let $J : U_{ad} \times \mathbb{R}^p)^3 \to \mathbb{R}$ be a continuously differentiable cost functional. Then the shape optimization problem reads as:

$$\left.\begin{array}{ll} \text{minimize} & J(\boldsymbol{\alpha}, \boldsymbol{y}) \\ \text{subj. to} & \mathbf{0} \in F(\boldsymbol{\alpha}, \boldsymbol{y}) + Q(\boldsymbol{y}) \\ & \boldsymbol{\alpha} \in U_{ad}. \end{array}\right\} \tag{$\mathbb{P}$}$$

In the sequel we shall assume that the assumptions of Theorem 1 are satisfied. Then ($\mathbb{P}$) is equivalent to the following nonlinear program:

$$\left.\begin{array}{ll} \text{minimize} & \mathcal{J}(\boldsymbol{\alpha}) := J(\boldsymbol{\alpha}, S(\boldsymbol{\alpha})) \\ \text{subj. to} & \boldsymbol{\alpha} \in U_{ad}, \end{array}\right\} \tag{$\tilde{\mathbb{P}}$}$$

which may be solved by algorithms of nonsmooth optimization (note, that $\mathcal{J}$ is locally Lipschitz continuous due to Theorem 1). Such algorithms, however, require knowledge of some subgradient information, usually in the form of one (arbitrary) subgradient from the Clarke subdifferential $\overline{\partial}\mathcal{J}$ at each iteration step. Following [2] and [4], we are not going to use Clarke's calculus (cf. [3]) to obtain the desired subgradient, but the substantially richer calculus developed by B. Mordukhovich. A straightforward application of this theory is the next result. For the rest of this section let $\bar{\boldsymbol{\alpha}} \in U_{ad}$ be arbitrary and put $\bar{\boldsymbol{y}} := S(\bar{\boldsymbol{\alpha}})$.

Lemma 1. $\overline{\partial}\mathcal{J}(\bar{\boldsymbol{\alpha}}) \subset \nabla_\alpha J(\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{y}}) + D^*S(\bar{\boldsymbol{\alpha}})(\nabla_y J(\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{y}}))$.

Therefore, it is sufficient to determine one element of the (limiting) coderivative $D^*S(\bar{\boldsymbol{\alpha}})$ $(\nabla_y J(\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{y}})) = \{\boldsymbol{v}^* \in \mathbb{R}^p \mid (\boldsymbol{v}^*, -\nabla_y J(\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{y}})) \in N_{\operatorname{Gr} S}(\bar{\boldsymbol{\alpha}})\}$, where $N_{\operatorname{Gr} S}$ stands for the (limiting) normal cone to the graph of S (cf. [5, 6]). To facilitate the computation of this quantity, we have the following result at hand:

Theorem 2. *For every $\boldsymbol{v}^* \in D^*S(\bar{\boldsymbol{\alpha}})(\nabla_y J(\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{y}}))$ there exists a vector $\boldsymbol{p}^* \in \mathbb{R}^p)^3$ such that $\boldsymbol{v}^* = \nabla_\alpha F(\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{y}})^T \boldsymbol{p}^*$ and $\boldsymbol{p}^*$ is a solution of the (limiting) adjoint GE:*

$$\mathbf{0} \in \nabla_y J(\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{y}}) + \nabla_y F(\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{y}})^T \boldsymbol{p}^* + D^*Q(\bar{\boldsymbol{y}}, -F(\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{y}}))(\boldsymbol{p}^*). \tag{AGE}$$

Proof. See Theorem 4.1 in [2]. □

In the rest of this section we show how one may express the coderivative D^*Q in terms of the data of the problem. First, we group the equations in (3) corresponding to each contact node, so that the multivalued part Q becomes: $Q(\boldsymbol{y}) = (\Phi(\boldsymbol{y}_1), \Phi(\boldsymbol{y}_2), \ldots, \Phi(\boldsymbol{y}_p))^T$, where $\boldsymbol{y}_i = \big((\boldsymbol{u}_\tau)_i, (\boldsymbol{u}_\nu)_i, \boldsymbol{\lambda}_i\big)^T \in \mathbb{R} \times \mathbb{R}_+ \times \mathbb{R}_+$ for every $i = 1, \ldots, p$ and

$$\Phi(\boldsymbol{a}) := \big(\mathcal{F}(|a_1|)a_3 \partial |a_1|,\ 0,\ N_{\mathbb{R}_+}(a_3)\big)^T \qquad \forall \boldsymbol{a} \in \mathbb{R} \times \mathbb{R}_+ \times \mathbb{R}_+.$$

Thus, for arbitrary $(\bar{\boldsymbol{y}}, \bar{\boldsymbol{q}}) \in \operatorname{Gr} Q$ and $\boldsymbol{d}^* \in (\mathbb{R}^3)^p$:

$$D^*Q(\bar{\boldsymbol{y}}, \bar{\boldsymbol{q}})(\boldsymbol{d}^*) = \begin{pmatrix} D^*\Phi(\bar{\boldsymbol{y}}_1, \bar{\boldsymbol{q}}_1)(\boldsymbol{d}_1^*) \\ D^*\Phi(\bar{\boldsymbol{y}}_2, \bar{\boldsymbol{q}}_2)(\boldsymbol{d}_2^*) \\ \vdots \\ D^*\Phi(\bar{\boldsymbol{y}}_p, \bar{\boldsymbol{q}}_p)(\boldsymbol{d}_p^*) \end{pmatrix}.$$

It means, that in order to obtain D^*Q, it is sufficient to evaluate the coderivative $D^*\Phi$ for every contact node. The computation of these quantities is facilitated by the natural decomposition

of $\operatorname{Gr}\Phi$ according to the corresponding contact and sliding modes (see Table 1; impossible combinations are crossed out):

$$\operatorname{Gr}\Phi = L \cup M_1 \cup M_2 \cup M_3^+ \cup M_3^- \cup M_4.$$

Due to the relatively simple structure of Φ it is already manageable for each one of these sets $\Sigma \in \{L, M_1, M_2, M_3^+, M_3^-, M_4\}$ above to express $N_{\operatorname{Gr}\Phi}(\bar{\boldsymbol{a}}, \bar{\boldsymbol{b}})$ for $(\bar{\boldsymbol{a}}, \bar{\boldsymbol{b}}) \in \Sigma$ exactly, in terms of the data of our problem. More importantly, no additional smoothness of $\mathcal{F}$ is required to carry out the analysis, except the one ensuring validity of Theorem 1. In addition, when $\mathcal{F}$ happens to be constant, one recovers the formulas in [2] for the two-dimensional case.

<table>
<tr><th></th><th>no contact:
$a_3 = 0,\ b_3 < 0$</th><th>weak contact:
$a_3 = 0,\ b_3 = 0$</th><th>strong contact:
$a_3 > 0,\ b_3 = 0$</th></tr>
<tr><td>sliding:
$a_1 \neq 0,$
$b_1 = \operatorname{sgn}(a_1)\mathcal{F}(\lvert a_1\rvert)a_3$</td><td rowspan="2">$L$</td><td>$M_2$</td><td>$M_1$</td></tr>
<tr><td>weak sticking:
$a_1 = 0,$
$\lvert b_1\rvert = \mathcal{F}(0)a_3$</td><td>$M_4$</td><td>$M_3^-$</td></tr>
<tr><td>strong sticking:
$a_1 = 0,$
$\lvert b_1\rvert < \mathcal{F}(0)a_3$</td><td>$\times\times\times$</td><td>$\times\times\times$</td><td>$M_3^+$</td></tr>
</table>

Table 1: Contact and sliding mode of $(\boldsymbol{a}, \boldsymbol{b}) \in \operatorname{Gr}\Phi$.

Acknowledgement: Financial support from the GAUK project no. 719912 of the Charles University is gratefully acknowledged.

References

[1] P. Beremlijski, J. Haslinger, M. Kočvara, J. V. Outrata: *Shape optimization in contact problems with Coulomb friction*. SIAM J. Opt. 13, 2002, pp. 561–587.

[2] P. Beremlijski, J. Haslinger, M. Kočvara, R. Kučera, J. V. Outrata: *Shape optimization in three-dimensional contact problems with Coulomb friction*. SIAM J. Opt. 20, 2009, pp. 416–444.

[3] F. F. Clarke: *Optimization and nonsmooth analysis*. John Wiley & Sons, New York, 1983.

[4] J. Haslinger, J. V. Outrata, R. Pathó: *Shape optimization in 2D contact problems with given friction and a solution-dependent coefficient of friction*. Set-Valued Var Anal 20, 2012, pp. 31–59.

[5] B. S. Mordukhovich: *Variational analysis and generalized differentiation, I: basic theory, II: applications*. Grundlehren Series (Fundamental Principles of Mathematical Sciences), Vols. 330 and 331, Springer-Verlag, Berlin-Heidelberg, 2006.

[6] R. T. Rockafellar, R. Wets: *Variational analysis*. Springer-Verlag, Berlin, 1998.

Shape optimization for Stokes problem with threshold slip

J. Haslinger, J. Stebel, T. Sassi

Faculty of Mathematics and Physics, Charles University in Prague
Institute of Mathematics AS CR, Prague
Laboratoire de Mathématiques Nicolas Oresme, Caen

1 Introduction

We study the Stokes system with a friction-type condition, which switches between a slip and no-slip stage depending on the magnitude of the shear stress. Our main goal is to study under which conditions concerning smoothness of the domain Ω, solutions to this problem depend continuously on variations of Ω. This is the basic property enabling us to prove the existence of optimal shapes for a large class of optimal shape design problems. In order to release the impermeability condition, whose numerical treatment could be troublesome, we use a penalty approach. We introduce a family of shape optimization problems with the penalized states and establish mutual relation between solutions to the original and the modified optimization problems when the penalty parameter tends to zero. Finally, we study a discretization of the penalized problem and its convergence properties.

2 Formulation of the problem

In this work we shall consider a specific family of domains, namely $\mathcal{O} = \{\Omega(\alpha)|\ \alpha \in \mathcal{U}_{ad}\}$, where

$$\Omega(\alpha) = \{(x_1, x_2)|\ x_1 \in (0,1),\ x_2 \in (\alpha(x_1), \gamma)\}, \tag{1}$$

$$\mathcal{U}_{ad} = \{\alpha \in C^{1,1}([0,1])|\ \alpha_{min} \le \alpha \le \alpha_{max} \text{ in } [0,1], |\alpha^{(j)}| \le C_j,\ j = 1,2 \text{ a.e. in } (0,1)\}, \tag{2}$$

see Figure 1. Here γ, α_{min}, α_{max}, C_1, C_2 are given positive constants chosen in such a way that $\mathcal{U}_{ad} \neq \emptyset$. The boundary $\partial\Omega(\alpha)$ is split into $S(\alpha)$ and $\Gamma(\alpha) = \partial\Omega(\alpha) \setminus \overline{S}(\alpha)$, where

$$S(\alpha) = \{(x_1, x_2)|\ x_1 \in (0,1),\ x_2 = \alpha(x_1)\},\ \alpha \in \mathcal{U}_{ad},$$

i.e. $S(\alpha)$ is the graph of α.

For any $\alpha \in \mathcal{U}_{ad}$ we consider the Stokes problem

$$-\Delta \boldsymbol{u} + \nabla p = \boldsymbol{f},\ \text{div } \boldsymbol{u} = 0 \text{ in } \Omega(\alpha) \tag{3a}$$

with the following boundary conditions:

$$\boldsymbol{u} = \boldsymbol{0} \qquad \text{on } \Gamma(\alpha), \tag{3b}$$

$$u_\nu = 0 \qquad \text{on } S(\alpha), \tag{3c}$$

$$\|\boldsymbol{\sigma}_\tau\| \le g \qquad \text{on } S(\alpha), \tag{3d}$$

$$\boldsymbol{u}_\tau \neq \boldsymbol{0} \Rightarrow \|\boldsymbol{\sigma}_\tau\| = g\ \&\ \exists \lambda \ge 0 : \boldsymbol{u}_\tau = -\lambda \boldsymbol{\sigma}_\tau \qquad \text{on } S(\alpha). \tag{3e}$$

Here $\boldsymbol{u} = (u_1, u_2)$ is the velocity field, p is the pressure and $\boldsymbol{f}$ is the external force. Further, $\boldsymbol{\nu}$, $\boldsymbol{\tau}$ denote the unit outward normal, and tangential vector to $\partial\Omega$, respectively. If $\boldsymbol{a} \in \mathbb{R}^2$ is

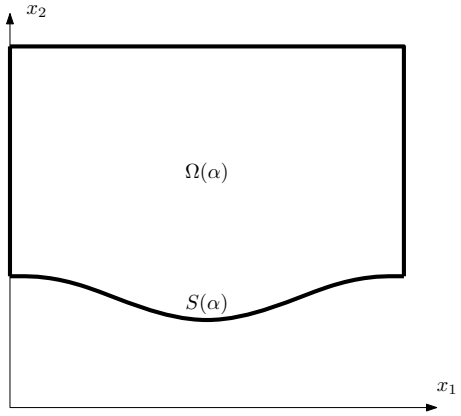


Figure 1: Geometry of the domain $\Omega(\alpha)$.

a vector then $a_\nu := \boldsymbol{a} \cdot \boldsymbol{\nu}$, $\boldsymbol{a}_\tau := \boldsymbol{a} - a_\nu \boldsymbol{\nu}$ is its normal component, and the tangential part on $\partial\Omega$, respectively. The Euclidean norm of $\boldsymbol{a}$ is denoted by $\|\boldsymbol{a}\|$. Finally, $\boldsymbol{\sigma}_\tau := \left(\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{\nu}}\right)_\tau$ stands for the shear stress and $g > 0$ a.e. on S is a given slip bound.

In what follows we shall suppose that $\boldsymbol{f} \in (L^2_{loc}(\mathbb{R}^2))^2$ and for simplicity of our analysis that g is a positive constant. It is known [1] that (3) has a unique weak solution $(\boldsymbol{u}(\alpha), p(\alpha))$. The weak formulation of (3) will be denoted by $(\mathcal{P}(\alpha))$ in the sequel. For the definition of the weak formulation and further details we refer to [3].

Finally, let $J : (\alpha, \boldsymbol{y}, q) \mapsto \mathbb{R}$ be the cost functional and denote $\mathfrak{J}(\alpha) := J(\alpha, \boldsymbol{u}(\alpha), p(\alpha))$. We shall study the following optimal shape design problem:

$$\left.\begin{array}{l} \textit{Find } \alpha^* \in \mathcal{U}_{ad} \textit{ such that} \\ \\ \forall \alpha \in \mathcal{U}_{ad} : \quad \mathfrak{J}(\alpha^*) \leq \mathfrak{J}(\alpha). \end{array}\right\} \tag{$\mathbb{P}$}$$

Our first result is the following theorem.

Theorem 1. *Let J be lower semicontinuous in the sense specified in [3], (2.9). Then $(\mathbb{P})$ has a solution.*

The proof strongly relies on the fact that the family $\mathcal{O}$ consists of domains with uniformly $C^{1,1}$-boundary. Note that for lower regularity of the boundaries such result cannot be expected.

3 Shape optimization with the penalized state problem

We propose a new shape optimization problem for the Stokes system with threshold slip with a penalization of the impermeability condition (3c). The boundary condition $\boldsymbol{u} \cdot \boldsymbol{\nu}^\alpha = 0$ on $S(\alpha)$ will be approximated by the following bilinear form:

$$c_\alpha(\boldsymbol{u}, \boldsymbol{v}) = \int_0^1 (\boldsymbol{u} \circ \alpha \cdot \boldsymbol{\nu}^\alpha)(\boldsymbol{v} \circ \alpha \cdot \boldsymbol{\nu}^\alpha) dx_1,$$

where $\boldsymbol{u} \circ \alpha \cdot \boldsymbol{\nu}^\alpha := \boldsymbol{u}(x_1, \alpha(x_1)) \cdot \boldsymbol{\nu}^\alpha(x_1)$, $x_1 \in (0, 1)$.

Let $\alpha \in \mathcal{U}_{ad}$ be fixed and $\varepsilon > 0$ be a penalty parameter. The penalized form of $(\mathcal{P}(\alpha))$ will be denoted by $(\mathcal{P}_\varepsilon(\alpha))$. Using the same technique as in [1] one can show that $(\mathcal{P}_\varepsilon(\alpha))$ has a unique solution $(\boldsymbol{u}_\varepsilon(\alpha), p_\varepsilon(\alpha))$ for any $\varepsilon > 0$.

Now we introduce the following family of shape optimization problems with the state problem $(\mathcal{P}_\varepsilon(\alpha))$. For any $\varepsilon > 0$ fixed, we define:

$$\left.\begin{array}{l} \textit{Find } \alpha_\varepsilon^* \in \mathcal{U}_{ad} \textit{ such that} \\ \\ \forall \alpha \in \mathcal{U}_{ad}: \quad \mathfrak{J}_\varepsilon(\alpha_\varepsilon^*) \le \mathfrak{J}_\varepsilon(\alpha), \end{array}\right\} \tag{$\mathbb{P}_\varepsilon$}$$

where $\mathfrak{J}_\varepsilon(\alpha) := J(\alpha, \boldsymbol{u}_\varepsilon(\alpha), p_\varepsilon(\alpha))$. One can prove the following result.

Theorem 2. *Under the assumption of Theorem 1, problem $(\mathbb{P}_\varepsilon)$ has a solution for any $\varepsilon > 0$.*

In the following theorem we establish the mutual relation between solutions of $(\mathbb{P})$ and $(\mathbb{P}_\varepsilon)$ for $\varepsilon \to 0+$.

Theorem 3. *Let J be lower semicontinuous and continuous in the sense specified in [3], (2.9) and (4.8), respectively. Then from any sequence $\{\alpha_\varepsilon^*\}$ of solutions to $(\mathbb{P}_\varepsilon)$, $\varepsilon \to 0+$ one can choose a subsequence (denoted by the same symbol) such that*

$$\alpha_\varepsilon^* \to \alpha^* \textit{ in } C^1([0,1]), \tag{4}$$

where α^* is a solution of $(\mathbb{P})$. Besides that, any accumulation point of $\{\alpha_\varepsilon^*\}$ in the sense of (4) has this property.

4 Approximation of $(\mathbb{P}_\varepsilon)$

In this section, we shall assume that $\varepsilon > 0$ is fixed. We introduce a finite element discretization of $(\mathcal{P}_\varepsilon(\alpha))$ and a discretization of the set $\mathcal{U}_{ad}$. Finally we will study convergence properties of such solutions if the discretization parameter $h \to 0+$.

4.1 Formulation of the discrete problem

Since for finite element methods it is convenient to use polygonal domains, we will consider piecewise linear approximations of $\mathcal{U}_{ad}$. On the other hand, as $\mathcal{U}_{ad}$ contains $C^{1,1}$-functions, this approximation of $\mathcal{U}_{ad}$ becomes external and some technical difficulties arise especially in the convergence analysis.

The set of discrete admissible shapes $\mathcal{U}_{ad}^h$ consists of continuous, piecewise linear functions on an equidistant partition of $[0,1]$ which satisfy constraints analogous to those ones imposed in (2), expressed in terms of difference quotients.

We will consider the system $\{\mathcal{T}_h(\alpha_h) |\ \alpha_h \in \mathcal{U}_{ad}^h\}$ which consists of topologically equivalent triangulations of $\overline{\Omega}(\alpha_h)$ (see e.g. [2]). The finite element discretization of the state problem is based on the Galerkin method with the discrete velocity and pressure spaces built on $\mathcal{T}_h(\alpha_h)$ and satisfying the Babuška-Brezzi condition (e.g. the Taylor-Hood finite element spaces). Consequently, the resulting discrete problem $(\mathcal{P}_{h\varepsilon}(\alpha_h))$ has a unique solution.

Analogously to the continuous setting, the discrete shape optimization problem is defined as the minimization of $\mathfrak{J}_{h\varepsilon}$ on $\mathcal{U}_{ad}^h$, where

$$\mathfrak{J}_{h\varepsilon}(\alpha_h) := J(\alpha_h, \boldsymbol{u}_{h\varepsilon}(\alpha_h), p_{h\varepsilon}(\alpha_h)).$$

with $(\boldsymbol{u}_{h\varepsilon}(\alpha_h), p_{h\varepsilon}(\alpha_h))$ being the solution of $(\mathcal{P}_{h\varepsilon}(\alpha_h))$. Thus, for each $\varepsilon > 0$ and $h > 0$, the discrete shape optimization problem reads:

$$\left.\begin{array}{l} \textit{Find } \alpha^*_{h\varepsilon} \in \mathcal{U}^h_{ad} \textit{ such that} \\ \\ \forall \alpha_h \in \mathcal{U}^h_{ad}: \quad \mathfrak{J}_{h\varepsilon}(\alpha^*_{h\varepsilon}) \leq \mathfrak{J}_{h\varepsilon}(\alpha_h). \end{array}\right\} \qquad (\mathbb{P}_{h\varepsilon})$$

The existence result for an optimal discrete shape is straightforward.

Theorem 4. *Let $h, \varepsilon > 0$ be fixed and $\mathfrak{J}_{h\varepsilon}$ be lower semicontinuous on $\mathcal{U}^h_{ad}$. Then $(\mathbb{P}_{h\varepsilon})$ has a solution.*

4.2 Convergence analysis

Finally, we establish the mutual relation between solutions to $(\mathbb{P}_{h\varepsilon})$ and $(\mathbb{P}_\varepsilon)$ as $h \to 0+$ keeping $\varepsilon > 0$ fixed, aiming to show that the discrete optimal shapes converge in some sense to an optimal shape of the continuous setting.

We have the following convergence result.

Theorem 5. *Let $\{\alpha^*_{h\varepsilon}\}$, $h \to 0+$ be a sequence of solutions to $(\mathbb{P}_{h\varepsilon})$, $h \to 0+$ and let J be continuous in the sense of [3], (5.7). Then there exists a subsequence of $\{\alpha^*_{h\varepsilon}\}$ (denoted by the same symbol) such that*

$$\alpha^*_{h\varepsilon} \to \alpha^*_\varepsilon \textit{ in } C([0,1]),$$

where α^*_ε is a solution of $(\mathbb{P}_\varepsilon)$.

5 Conclusion

The contribution was devoted to the shape optimization of the Stokes problem with threshold slip boundary condition. We have shown the existence of an optimal shape, the relation to a penalized shape optimization problem which releases the impermeability condition and finally, we studied convergence properties of a discretization of the penalized problem.

Acknowledgement: The research of the first author was supported by the grant P201/12/0671 of GAČR. The second author acknowledges the support of the grant 201/09/0917 of GAČR and RVO: 67985840. Finally a part of this paper was done in co-operation of the first and the third author in the frame of ERASMUS project.

References

[1] H. Fujita: *A mathematical analysis of motions of viscous incompressible fluid under leak or slip boundary conditions.* RIMS Kôkyûroku 888, 1994, pp. 199–216.

[2] J. Haslinger, R. Mäkinen: *Introduction to shape optimization: theory, approximation, and computation.* Advances in Design and Control, DC07. Society for Industrial Mathematics, 2003.

[3] J. Haslinger, J. Stebel, T. Sassi: *Shape optimization for Stokes problem with threshold slip.* Appl. Math., 2012, submitted.

Numerical solution of the discrete barrier option pricing problem

J. Hozman

Technical University of Liberec

1 Introduction

During the last decade, financial models have acquired increasing popularity in option pricing. The valuation of different types of option contracts is very important in modern financial theory and practice, especially exotic options have become very popular speculation instruments in recent years. The problem of determining the fair price of such an option is standardly formulated in the well-known Black–Scholes equation, firstly presented in [3].

A huge amount of literature has been devoted to the solving of this equation or its modification. The performance demands on the valuation process are very high in this case. Moreover, most of the analytical formulas for these options is limited by strong assumptions, which led to the application of numerical methods instead. Therefore, the main goal of this paper is to develop an efficient, robust and accurate method for the exotic option pricing problem, which arises from the concept of the discontinuous Galerkin (DG) approach (cf. [2, 4, 7]) and enables better resolving of occurred special properties of certain types of exotic options, in comparison with the standard finite element approach, see e.g. [1, 6, 8] and the references cited therein.

2 Discrete barrier option pricing problem

In this paper, we focus only on one family of exotic options such as discrete barrier options. Furthermore, we shall concentrate only on a discrete double time–independent barrier knock-out option, i.e. an option that expires worthless if one of the two barriers has been hit at a monitoring date, for more details see [1, 8]. Let $M := \{0 = t_0^M < t_1^M < \ldots < t_{l-1}^M < t_l^M = T\}$ be the set of monitoring dates and B_- be the lower barrier and B_+ the upper barrier active only at discrete instances $t_l^M \in M$.

Let $\Omega := (S_{min}, S_{max})$, $0 < S_{min} < B_- < B_+ < S_{max}$, be a bounded open interval and T stands for the maturity. We denote by x the price of an underlying asset (e.g. stock) and by t the time to expiry of the option. The price $u : Q_T := \Omega \times (0,T) \to \mathbb{R}$ of the discrete barrier option satisfies the Black–Scholes partial differential equation with initial and boundary conditions, i.e.

$$\frac{\partial}{\partial t}u(x,t) - \frac{1}{2}\sigma^2 x^2 \frac{\partial^2}{\partial x^2}u(x,t) - rx\frac{\partial}{\partial x}u(x,t) + ru(x,t) = 0 \quad \text{in } Q_T, \tag{1}$$

$$u(S_{min},t) = 0 \quad \text{and} \quad u(S_{max},t) = 0, \tag{2}$$

$$u(x,0) = \begin{cases} \max(x-K,0)\cdot\chi_{[B_-,B_+]}, & \text{(call)} \\ \max(K-x,0)\cdot\chi_{[B_-,B_+]}, & \text{(put)} \end{cases}, \quad x \in \Omega, \tag{3}$$

where $\sigma > 0$ and $r > 0$ are model parameters denoting the volatility of stock price and the risk–free interest rate, respectively. In real markets, values r and σ vary with time, but to keep the model and analysis simple, we assume r and σ to be constant.

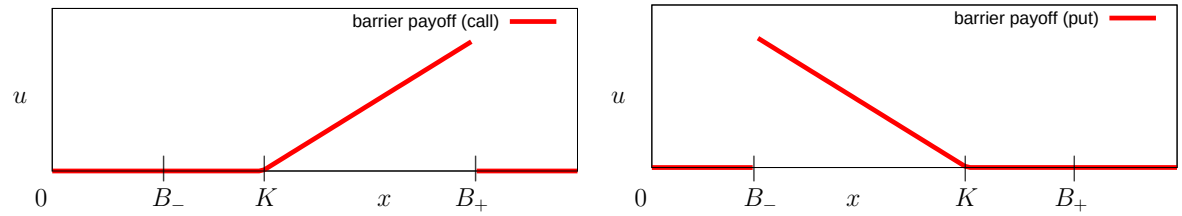


Figure 1: Initial values of a double discrete barrier knock-out call (left) and a double discrete barrier knock-out put (right) with strike price K.

From the mathematical point of view the problem (1)–(3) represents a convection–diffusion-reaction equation equipped with a set of two homogeneous Dirichlet boundary conditions (2) prescribed at the endpoints of Ω and with the initial condition (3), where the symbol K stands for the strike price and $\chi_{[B_-,B_+]}$ denotes the characteristic function of the barrier interval.

Moreover the discrete monitoring of the contract introduces an updating of the solution $u(x,t)$ at the monitoring dates $t_l^M \in M$, i.e.

$$u(x,t_l^M) = \lim_{\varepsilon \to 0+} u(x,t_l^M - \varepsilon) \cdot \chi_{[B_-,B_+]}. \tag{4}$$

The knock-out clause (4) at monitoring instances introduces a discontinuity at the barriers, as illustrated in Figure 1 for the first monitoring date.

3 DG discretization

The discontinuous Galerkin approach is suitable for problems with irregular solutions, because its framework originally arises from a generally discontinuous piecewise polynomial approximation $u_h(t)$ describing the global solution $u(x,t)$ on the whole domain Ω, i.e.

$$u_h(t) \in S_h = \left\{ v_h \in L^2(\Omega);\ v_h\big|_I \in P^p(I)\ \forall\, I \in \mathcal{T}_h \right\} \subset H^1(\Omega) \tag{5}$$

where $\mathcal{T}_h$ is a family of partitions of the closure $\overline{\Omega} = [S_{min}, S_{max}]$ into closed mutually disjoint subintervals I, and $P^p(I)$ denotes the space of all polynomials of degree $\leq p$ on element I.

In order to obtain a space semi–discrete DG scheme from [7], we multiply (1) by a test function $v_h \in S_h$, integrate over an element $I \in \mathcal{T}_h$ and use integration by parts in the diffusion and convection terms of (1) subsequently. Further, we sum over all $I \in \mathcal{T}_h$ and add some artificial terms vanishing for the exact solution such as penalty and stabilization terms, which replace the inter–element discontinuities and guarantee the stability of the resulting numerical scheme, respectively. Consequently, we employ a concept of an upwind numerical flux (see [5]) for the discretization of the convection term and end up with the following DG formulation for the semi–discrete solution $u_h(t)$ represented by a system of ordinary differential equations, i.e.

$$\frac{d}{dt}\left(u_h(t), v_h\right) + \mathcal{A}_h(u_h(t), v_h) = 0 \quad \forall\, v_h \in S_h,\ \forall\, t \in (0,T) \tag{6}$$

where a form $\mathcal{A}_h(\cdot,\cdot)$ stands for the semi–discrete variant of the linear differential operator in (1), see [7].

In order to obtain the discrete solution, it is necessary to equip the scheme (6) with suitable solvers for the time integration. The suggested implicit time discretization is suitable for avoiding

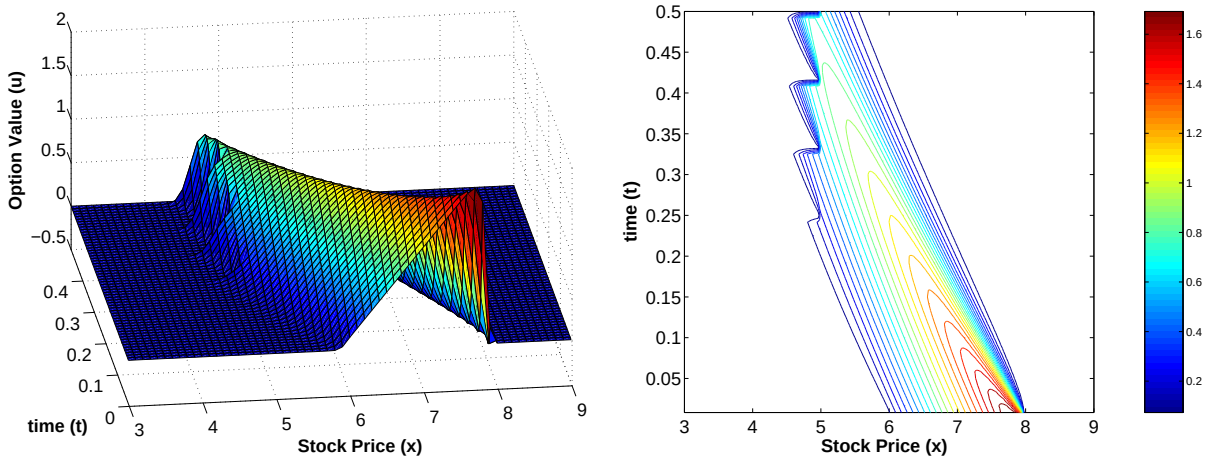


Figure 2: The 3D plot of value function of discrete barrier put option (left) and the corresponding isolines of the option price (right).

the strong time step restriction of explicit time schemes. Moreover, a bilinearity of the form $\mathcal{A}_h(\cdot,\cdot)$ directly implies that the used implicit treatment in (6) corresponds to a system of linear algebraic equations without employing any additional linearisation, cf. [1, 8].

For the sake of clarity, we use the simplest implicit method — backward Euler method — for the time discretization and introduce the fully discrete scheme. We now partition $[0,T]$ as $0 = t_0 < t_1 < t_2 < \ldots < t_N = T$, denoting each time step by $\tau_l = t_l - t_{l-1}$. We compute the approximate values u_h^l of the exact solution $u(t_l)$ only at given time levels t_l according the following formula, i.e.

$$\left(u_h^l, v_h\right) + \tau_l \mathcal{A}_h\left(u_h^l, v_h\right) = \left(u_h^{l-1}, v_h\right) \quad \forall\, v_h \in S_h, \quad l = 1, 2, \ldots, N \tag{7}$$

with initial state u_h^0 as S_h-approximation of (3) and monitoring constraints $u_h^l := u_h^l \cdot \chi_{[B_-,B_+]}$ valid only at monitoring dates $t_l^M \in M$. Finally, the system (7) is then solved by a suitable linear algebraic solver.

4 Results and conclusion

In order to illustrate the potency of the derived numerical scheme (7) for a solution of discrete barrier options, we consider the knock-out call option with the expiration date $T = \frac{6}{12}$ (e.g. 6 months) and the strike price $K = 6.0$. The prescribed barriers are $B_- = 5.0$, $B_+ = 8.0$ and the computational domain was set as $\Omega = (3, 9)$. The Black–Scholes model parameters were the risk–free interest rate $r = 0.9y^{-1}$ and the volatility $\sigma^2 = 10^{-6}y^{-1}$. We carried out computations by piecewise cubic approximations on a priori uniformly adapted partition of Ω with constant time step $\tau = \frac{1}{120}$, used the restarted GMRES for the solving of linear systems and considered monthly monitoring.

Since $\sigma^2 \ll r$, the convection term is large compared to the diffusive term and the problem is said to be convection dominated and the partial differential equation exhibits a hyperbolic behaviour, i.e. the first-order hyperbolic term involving $\frac{\partial u}{\partial x}$ propagates information in the approximation solution from the right to the left of the x-axis, as illustrated in Figure 2 (left) together with

the corresponding isolines of the option price in the space–time plot with well-resolved monthly monitoring constraints, see Figure 2 (right).

We have dealt with the numerical solution of the discrete barrier option pricing model, represented by the linear convection–diffusion–reaction equation. We have derived the above mentioned numerical scheme: from the continuous problem, over the semi–discrete one to the fully discrete one. The whole method is based on the space semi–discretization by the discontinuous Galerkin method in space and on the implicit Euler method used for discretization in time. For the future work, we intend to extend this concept to a simple theoretical analysis and also to the multivariate Black–Scholes equation describing basket options.

References

[1] Y. Achdou, O. Pironneau: *Computational methods for option pricing*. Philadelphia, SIAM, 2005.

[2] D. N. Arnold, F. Brezzi, B. Cockburn, L. D. Marini: *Unified analysis of discontinuous Galerkin methods for elliptic problems*. SIAM J. Numer. Anal. 39 (5), 2002, pp. 1749–1779.

[3] F. Black, M. Scholes: *The pricing of options and corporate liabilities*. J. Political Economy 81, 1973, pp. 637–659.

[4] B. Cockburn, G. E. Karniadakis, C.–W. Shu, editors: *Discontinuous Galerkin methods*. Springer, Berlin, 2000.

[5] M. Feistauer, J. Felcman, I. Straškraba: *Mathematical and computational methods for compressible flow*. Oxford University Press, Oxford, 2003.

[6] G. Fusai, S. Sanfelici, A. Tagliani: *Practical problems in the numerical solution of PDE's in finance*. Rend. Studi Econ. Quant 2001, 2002, pp. 105–132.

[7] B. Riviére: *Discontinuous Galerkin methods for solving elliptic and parabolic equations: theory and implementation*. Frontiers in Applied Mathematics. Society for Industrial and Applied Mathematics, Philadelphia, 2008.

[8] R. Seydel: *Tools for computational finance:* 4^{th} *edition*. Springer, Berlin, 2008.

Classification of zeros of quaternionic polynomials

D. Janovská, G. Opfer

Institute of Chemical Technology, Prague
University of Hamburg

1 Introduction

Polynomials with quaternionic coefficients located on only one side of the powers (we call them simple polynomials) may have two different types of zeros: isolated and spherical zeros. We will give a characterization of these types. The main tool is the representation of the powers of a quaternion as a real, linear combination of the quaternion and the number one.

In the two–sided polynomials the coefficients are located at both sides of the powers. We show that in this case there are, in addition, three more classes of zeros defined by the rank of a certain real 4×4 matrix. The essential tool is the description of the polynomial p by a matrix equation $P(z) := \mathbf{A}(z)z + B(z)$, where $\mathbf{A}(z)$ is a real 4×4 matrix determined by the coefficients of the given polynomial p and P, z, B are real column vectors with four rows.

2 Preliminaries

By $\mathbb{R}$, $\mathbb{C}$ we denote the fields of real and complex numbers, respectively, and by $\mathbb{Z}$ the set of integers. By $\mathbb{H}$ we denote the skew field of quaternions.

Let $\mathbb{H} = \mathbb{R}^4$ be equipped with the ordinary vector space structure with an additional multiplicative operation $\mathbb{H} \times \mathbb{H} \longrightarrow \mathbb{H}$ which most easily can be defined by a multiplication of the four basis elements

$$(1,0,0,0) = \mathbf{1}, \quad (0,1,0,0) = \mathbf{i}, \quad (0,0,1,0) = \mathbf{j}, \quad (0,0,0,1) = \mathbf{k} \; :$$

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -\mathbf{1}\,. \tag{1}$$

An element $x = (x_1, x_2, x_3, x_4) \in \mathbb{H}$, $x_1, x_2, x_3, x_4 \in \mathbb{R}$, has the representation

$$x = \; x_1\mathbf{1} + x_2\mathbf{i} + x_3\,\mathbf{j} + x_4\mathbf{k},$$

If we denote $\mathbf{v} = (x_2, x_3, x_4) \in \mathbb{R}^3$ the vector part of x then, the quaternion x has the representation:

$$x = (x_1, \mathbf{v}), \quad x_1 \in \mathbb{R}, \quad \mathbf{v} \in \mathbb{R}^3.$$

For $x = (x_1, x_2, x_3, x_4) = (x_1, \mathbf{v}) \in \mathbb{H}$, $y = (y_1, y_2, y_3, y_4) = (y_1, \mathbf{w}) \in \mathbb{H}$ it follows from (1) that

$$\begin{aligned} x\,y \;\; &= \;\; (x_1y_1 - x_2y_2 - x_3y_3 - x_4y_4)\,\mathbf{1} + (x_1y_2 + x_2y_1 + x_3y_4 - x_4y_3)\,\mathbf{i} \\ &\quad + (x_1y_3 - x_2y_4 + x_3y_1 + x_4y_2)\,\mathbf{j} + (x_1y_4 + x_2y_3 - x_3y_2 + x_4y_1)\,\mathbf{k} \\ &= \;\; (x_1y_1 - \mathbf{v}\cdot\mathbf{w},\; x_1\mathbf{w} + y_1\mathbf{v} + \mathbf{v}\times\mathbf{w}), \end{aligned} \tag{2}$$

where $\cdot$, $\times$ are the dot and vector products in $\mathbb{R}^3$, respectively. Obviously, in general, the multiplication is not commutative. Given $x = (x_1, x_2, x_3, x_4) \in \mathbb{H}$, the conjugate $\overline{x}$ of x is defined to be

$$\overline{x} = (x_1, -x_2, -x_3, -x_4). = \Re\, x - \mathrm{Vec}\, x.$$

We define the absolute value of x by

$$|x| = \sqrt{x_1^2 + x_2^2 + x_3^2 + x_4^2}. \tag{3}$$

The space $\mathbb{H}$ is a normed vector space over $\mathbb{H}$, where the norm is introduced in (3).

Example Let us see a small example. Let $p_2(z) = z^2 + 1$. This quadratic polynomial has no real zero and it has two imaginary zeros $z_{1,2} = \pm\mathbf{i}$. How many zeros it has as a quadratic quaternionic polynomial? Let $z = h^{-1}z_{1,2}h$, where $h \in \mathbb{H} \setminus \{0\}$ is arbitrary. Then

$$z^2 + 1 = h^{-1}z_{1,2}h\, h^{-1}z_{1,2}h + 1 = h^{-1}\mathbf{i}^2 h + 1 = 0\,.$$

As a quadratic quaternionic polynomial, p_2 has infinitely many zeros.

Definition Two quaternions $a, b \in \mathbb{H}$ are called equivalent, denoted by $a \sim b$, if

$$a \sim b \quad \Longleftrightarrow \quad \exists\, h \in \mathbb{H} \setminus \{0\} \quad \text{such that} \quad a = h^{-1}bh. \tag{4}$$

The set

$$[a] := \left\{u \in \mathbb{H} : u = h^{-1}ah \quad \text{for all} \quad h \in \mathbb{H} \setminus \{0\}\right\} \tag{5}$$

will be called an equivalence class of a.

The relation $\sim$ is indeed an equivalence relation. If a is not real, then $[a]$ always contains infinitely many elements,

$$[a] = \{z \in \mathbb{H} :\ \Re z = \Re a, \quad \text{and} \quad |z| = |a|\}\,, \tag{6}$$

and the equivalence class $[a]$ can be regarded as a two dimensional sphere in $\mathbb{R}^4$.

Let $z := (z_1, z_2, z_3, z_4) \in \mathbb{H}$. Then it follows from (6) that $\overline{z} \in [z]$. If $z \in \mathbb{H}$ is not real then the equivalence class $[z]$ contains exactly two complex numbers $a \in \mathbb{C}$ and $\overline{a} \in \mathbb{C}$ where

$$a = (z_1, +\sqrt{z_2^2 + z_3^2 + z_4^2}, 0, 0) = z_1 + |\text{Vec z}|\mathbf{i} \in [\text{z}],$$

i.e., a is the only complex element in $[z]$ with a non negative imaginary part. The complex number a will be called the complex representative of $[z]$.

We introduce a mapping $\omega_1 : \mathbb{H} \longrightarrow \mathbb{R}^{4\times 4}$ by

$$\omega_1(a) := \begin{pmatrix} a_1 & -a_2 & -a_3 & -a_4 \\ a_2 & a_1 & -a_4 & a_3 \\ a_3 & a_4 & a_1 & -a_2 \\ a_4 & -a_3 & a_2 & a_1 \end{pmatrix} \in \mathbb{R}^{4\times 4},. \tag{7}$$

The mapping ω_1 represents the isomorphic image of a quaternion $a = (a_1, a_2, a_3, a_4)$ in the matrix space $\mathbb{R}^{4\times 4}$. Thus we have

$$\omega_1(ab) = \omega_1(a)\omega_1(b).$$

For $a := (a_1, a_2, a_3, a_4) \in \mathbb{H}$, we define a column operator $\text{col} : \mathbb{H} \to \mathbb{R}^4$ by $\text{col}(a) := \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{pmatrix}$.

This column operator enables us to regard a quaternion as a matrix with one column and four rows.

Let $\mathbf{A}$ be a square matrix over $\mathbb{C}$ of order n. Then, see e.g. Horn & Johnson, [1], any power $\mathbf{A}^j$ belongs to a linear hull of the powers of the matrix $\mathbf{A}$ up to the degree ν of the minimal polynomial of $\mathbf{A}$:

$$\mathbf{A}^j \in \langle \mathbf{I},\, \mathbf{A},\, \mathbf{A}^2,\, \ldots\,, \mathbf{A}^{\nu-1} \rangle\,,\ j \in \mathbb{N}\,.$$

We will apply this theory to the real matrix $\omega_1(a)$ that represents the quaternion a. It has the minimal polynomial

$$\mu(\omega(a)) = \lambda^2 - 2\lambda a_1 + |a|^2 \quad \text{i.e.} \quad \nu = 2\,.$$

As a consequence, all powers $z^j, j \in \mathbb{Z}$, of a quaternion z have the form $z^j = \alpha z + \beta$ with real α, β. In order to determine the numbers α, β we set up the following iteration

$$\begin{aligned} z^j &= \alpha_j z + \beta_j, \quad \alpha_j, \beta_j \in \mathbb{R}, \quad j = 0, 1, \ldots, \text{ where} \\ \alpha_0 &= 0, \quad \beta_0 = 1, \\ \alpha_{j+1} &= 2\Re z\, \alpha_j + \beta_j, \\ \beta_{j+1} &= -|z|^2 \alpha_j, \quad j = 0, 1, \ldots \end{aligned} \tag{8}$$

3 Simple (one–sided) quaternionic polynomials

Let $p_n(z)$ be a given polynomial of degree n, n positive integer,

$$p_n(z) = \sum_{j=0}^{n} a_j z^j\,, \quad z,\, a_j \in \mathbb{H},\ j = 0, 1, 2, \ldots, n, \quad a_0,\ a_n \neq 0\,. \tag{9}$$

Polynomial $p_n(z)$ in (9) is called one-sided (or simple) quaternionic polynomial.

The set of zeros of the polynomial of type (9) will separate into two classes:

Definition Let z_0 be a zero of a simple quaternionic polynomial (9).

- If z_0 is not real and has the property that $p_n(z) = 0$ for all $z \in [z_0]$, then we will say that z_0 is a spherical zero.
- If z_0 is real or does not generate a spherical zero, it is called an isolated zero.
- The number of zeros of p_n is defined as the number of equivalence classes, which contain at least one zero of p_n.

By means of (8) the polynomial p_n can be written as

$$p_n(z) := \sum_{j=0}^{n} a_j z^j = \sum_{j=0}^{n} a_j(\alpha_j\, z + \beta_j) = \left(\sum_{j=0}^{n} \alpha_j a_j \right) z + \sum_{j=0}^{n} \beta_j a_j =: A(z)z + B(z).$$

We have the following classification of the zeros z_0 of p_n given in (9):

(i) z_0 is real. By definition, z_0 is isolated.

(ii) z_0 is not real:

- $A(z_0) = 0 \Rightarrow z_0$ is spherical, all $z \in [z_0]$ are zeros of p_n.
- $A(z_0) \neq 0 \Rightarrow z_0$ is isolated.

The computation of all zeros of p_n, including their types, can be reduced to the computation of all zeros of a real polynomial of degree $2n$. For details, see [2].

4 Two–sided quaternionic polynomials

The two–sided quaternionic polynomial has the form

$$p\,(z) := \sum_{j=0}^{n} a_j z^j b_j,\ z,\ a_j,\ b_j \in \mathbb{H},\ j = 0, 1, \ldots, n \in \mathbb{N}\,,\ a_0 b_0 \neq 0,\ a_n b_n \neq 0\,. \tag{10}$$

By means of (8), the two–sided quaternionic polynomial p can be written as

$$p\,(z) \ := \ \sum_{j=0}^{n} a_j z^j b_j = \sum_{j=0}^{n} a_j(\alpha_j\, z + \beta_j) b_j = C(z) + B(z), \quad \text{where} \tag{11}$$

$$C(z) \ := \ \sum_{j=0}^{n} \alpha_j a_j\, z\, b_j, \qquad B(z) := \sum_{j=0}^{n} \beta_j a_j b_j\,. \tag{12}$$

Moreover, if we apply the operator col to the equations (11) to (12)we can rewrite the equation $p(z) = 0$ in the equivalent form

$$P(z) := \mathrm{col(p(z))} = \mathbf{A}(\mathrm{z})\mathrm{col(z)} + \mathrm{col(B(z))} = \mathrm{col}(0) := \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{13}$$

From these results we obtain a classification of the zeros of two-sided quaternionic polynomial p as follows:

Definition Let z be a zero of p, defined in (10), and let $z_0 \in [z]$ be the complex representative of $[z]$.
The zero z will be called zero of type k if $\mathrm{rank}(\mathbf{A}(\mathrm{z}_0)) = 4 - \mathrm{k},\ 0 \leq \mathrm{k} \leq 4$.

A zero of type 4 ($\mathrm{rank}(\mathbf{A}(\mathrm{z}_0)) = 0$) will be called the spherical zero. It has the property that all $z \in [z_0]$ are zeros. A zero of type 0 will be called isolated zero. In this case $z = -(\mathbf{A}(z_0))^{-1}\mathrm{col(B(z_0))}$ is the only zero in $[z_0]$. We will also call a real zero an isolated zero. For details see [3].

5 Number of zeros of the quaternionic polynomials

Definition Let p be any quaternionic polynomial of degree $n \geq 2$. By $\#Z(p)$ we understand the number of equivalence classes in $\mathbb{H}$ which contain zeros of p. We call this number, essential number of zeros of p.

By this definition, $p(z) := z^2 + 1$ has one essential zero, since $\mathbf{i}$ and $-\mathbf{i}$ are located in the same equivalence class.

All polynomials with real coefficients and degree n as well as all quaternionic, one-sided polynomials of degree n have at most n essential zeros, see [3]. The essential number $\#Z(p)$ of zeros of the two–sided quaternionic polynomial of degree n is, in general, not bounded by n. Our conjecture is that in this case the essential number will not exceed $2n$.

References

[1] R. A. Horn, C. R. Johnson: *Matrix analysis*. Cambridge University Press, Cambridge, 1992.

[2] D. Janovská, G. Opfer: *A note on the computation of all zeros of simple quaternionic polynomials*. SIAM J. Numer. Anal. 48, 2010, pp. 244–256.

[3] D. Janovská, G. Opfer: *The classification and the computation of the zeros of quaternionic, two-sided polynomials*. Numerische Mathematik 115 (1), 2010, pp. 81–100.

Design of an object oriented framework for algebraic multigrid

P. Jiránek, S. Gratton, X. Vasseur, P. Hénon

[1,2,3] CERFACS, Toulouse, France
[4] Total SA, Centre Scientifique et Technique Jean-Féger, Pau

Multigrid methods are among the most efficient methods for solving and preconditiong of discretized partial differential equations. They employ the effects of relaxation and coarse-grid correction in a recursive way leading to a method with the computational cost depending linearly on the problem size. While the components of the "classical" geometric multigrid are constructed using the information involving the geometry and the nature of the problem to be solved, the algebraic multigrid (AMG) is defined entirely by the information contained in the given linear algebraic system and in a sense can be used as a "black-box" method. This makes the use of AMG attractive in particular for solving problems on complicated domains and unstructured grids, where the geometric multigrid can be hardly used if its application would be even possible.

Application of AMG splits in the setup and the solution part. In the setup part, the multigrid hierarchy of levels is created by recursively applying a procedure, which from a given input "fine" level constructs an output "coarse" level, until a certain stopping criterion is satisfied (e.g., the coarsest grid is small enough). In most AMG methods, common patterns can be found in the setup part. First, for each grid point a set of strongly coupled neighbors is determined based on certain criteria applied to the entries of the matrix associated with the given level. Using this information, the coarsening is then constructed. It describes how the coarse grid is created from the input fine grid (e.g., splitting the input grid to the sets coarse points and fine points in the classical AMG or partitioning to aggregates in aggregation-based AMG). Finally, the transfer operators can be defined from the given coarsening.

These common patterns in AMG motivate us to create a framework which involves some attractive features of modern object oriented languages like abstraction, polymorphism, and generic programming, and allows to implement various AMG algorithms and their components in a unified manner. We base our package on the Trilinos library (`http://trilinos.sandia.gov`), in particular on the packages Epetra for the basic communication and algebraic "core" and Teuchos. Our goal is to create an AMG package which would allow namely to:

- realize any kind of AMG method including the classical and aggregation-based AMG (even combining them in one multigrid hierarchy),
- combine various AMG components or to implement new custom ones from scratch,
- recompute already constructed multigrid hierarchy by reusing some of its previously computed parts.

In the presentation, we recall some basic AMG algorithms and describe the design of our framework including its current state of development. We illustrate its use on some academic numerical experiments including experiments on problems arising in reservoir simulations.

FETI method in civil engineering problems

J. Kruis, T. Koudelka

Faculty of Civil Engineering, Czech Technical University in Prague

Dedicated to Professor Ivo Marek in ocassion of his 80th birthday.

1 Introduction

This contribution deals with application of methods of domain decomposition to problems with imperfect bond on material interfaces or slip of soil along slip surface. Especially, the FETI method is used because it defines nodal unknowns on subdomains independently on other subdomains. The continuity across the subdomain interfaces is enforced by Lagrange multipliers, therefore at least two displacements are stored in the same point on subdomain interface. In the classical FETI method, these displacements are enforced to be equal but in generalised approach, different values can be enforced.

The continuity condition is replaced by slip condition based on bond slip law. The new condition generates an additional vector or matrix in the coarse problem. Complicated laws with softening is solved iteratively with the help of stiffness reduction.

2 Brief overview of FETI method

Finite element tearing and interconnecting method is a nonoverlapping domain decomposition method which transforms an original problem to the dual one which is solved by modified conjugate gradient method. Coarse space based on rigid body modes is used for fast exchange of informations during the iteration process. Overview of the method and many applications of the method can be found in references [1] and [2].

Let a domain Ω with boundary Γ is decomposed into m nonoverlapping subdomains Ω_j with boundaries Γ_j. Let the problem solved contain continuous unknown vector function $\boldsymbol{u}(\boldsymbol{x})$ which depends on the spatial coordinates $\boldsymbol{x}$. The unknown function is approximated by the finite element method and vector of unknown nodal values is denoted $\boldsymbol{d}$. In more detail, unknown nodal values on the j-th subdomain are collected in the vector $\boldsymbol{d}_j$. Because of decomposition, each subdomain contains its own unknowns on interface. The interface unknowns have to satisfy the continuity condition because the original problem is continuous. If a new vector of all nodal unknowns is defined in the form

$$\boldsymbol{d}^T = \left(\boldsymbol{d}_1^T, \boldsymbol{d}_2^T, \ldots, \boldsymbol{d}_m\right) \tag{1}$$

and a new matrix

$$\boldsymbol{B} = (\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_m) \tag{2}$$

is assembled, the continuity condition along the whole interface can be written

$$\boldsymbol{B}\boldsymbol{d} = \boldsymbol{0} \tag{3}$$

For each subdomain, a system of algebraic equations can be defined in the form

$$\boldsymbol{K}_j \boldsymbol{d}_j = \boldsymbol{f}_j - \boldsymbol{B}_j^T \boldsymbol{\lambda} \tag{4}$$

where $\boldsymbol{K}_j$ denotes the subdomain matrix, $\boldsymbol{f}_j$ denotes the subdomain right hand side vector. The stiffness matrix of the whole problem has the form

$$\boldsymbol{K} = \begin{pmatrix} \boldsymbol{K}_1 & & & \boldsymbol{0} \\ & \boldsymbol{K}_2 & & \\ & & \ddots & \\ \boldsymbol{0} & & & \boldsymbol{K}_m \end{pmatrix} \tag{5}$$

and the total right hand side vector can be written

$$\boldsymbol{f}^T = (\boldsymbol{f}_1^T, \boldsymbol{f}_2^T, \ldots, \boldsymbol{f}_m^T) \tag{6}$$

System of all equations has the form

$$\boldsymbol{K}\boldsymbol{d} = \boldsymbol{f} - \boldsymbol{B}^T \boldsymbol{\lambda} \tag{7}$$

and it is accompanied with the continuity conditions (3).

The vector of unknown nodal values can be expressed from the relationship (7) in the form

$$\boldsymbol{d} = \boldsymbol{K}^+ (\boldsymbol{f} - \boldsymbol{B}^T \boldsymbol{\lambda}) + \boldsymbol{R}\boldsymbol{\alpha} \tag{8}$$

where $\boldsymbol{K}^+$ is the pseudoinverse matrix, the matrix $\boldsymbol{R}$ contains basis vectors of kernel of matrices $\boldsymbol{K}_j$ which are denoted $\boldsymbol{R}_j$. The matrix $\boldsymbol{R}$ has the form

$$\boldsymbol{R} = \begin{pmatrix} \boldsymbol{R}_1 & & & \boldsymbol{0} \\ & \boldsymbol{R}_2 & & \\ & & \ddots & \\ \boldsymbol{0} & & & \boldsymbol{R}_m \end{pmatrix} \tag{9}$$

If a matrix $\boldsymbol{K}_j$ is nonsingular, the kernel contains no basis vector and the matrix $\boldsymbol{R}_j$ is removed from the matrix (9). The vector $\boldsymbol{f} - \boldsymbol{B}^T \boldsymbol{\lambda}$ has to be orthogonal to the kernels. The orthogonality can be written in the form

$$\boldsymbol{R}^T (\boldsymbol{f} - \boldsymbol{B}^T \boldsymbol{\lambda}) = \boldsymbol{0} \tag{10}$$

Substitution of expression (8) to continuity condition (3) results to

$$\boldsymbol{B}\boldsymbol{K}^+ \boldsymbol{f} - \boldsymbol{B}\boldsymbol{K}^+ \boldsymbol{B}^T \boldsymbol{\lambda} + \boldsymbol{B}\boldsymbol{R}\boldsymbol{\alpha} = \boldsymbol{0} \tag{11}$$

The previous equation together with the solvability condition (10) can be written in the form

$$\begin{pmatrix} \boldsymbol{B}\boldsymbol{K}^+ \boldsymbol{B}^T & -\boldsymbol{B}\boldsymbol{R} \\ -\boldsymbol{R}^T \boldsymbol{B}^T & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\alpha} \end{pmatrix} = \begin{pmatrix} \boldsymbol{B}\boldsymbol{K}^+ \boldsymbol{f} \\ -\boldsymbol{R}^T \boldsymbol{f} \end{pmatrix} \tag{12}$$

3 Modification of FETI method

Modification of the FETI method for problems dealing with perfect or imperfect bond was introduced in reference [3]. The continuity condition (3) is associated with the perfect bond where no cracks and other inelastic effects occur. On the other hand, in the case of inelastic behaviour, the continuity condition should be replaced by interface condition which deals with the interface because cracks or slips can occur and the continuity is violated.

The interface condition can be expressed in the general form

$$\boldsymbol{B}\boldsymbol{d} = \boldsymbol{c} \tag{13}$$

where $\boldsymbol{c}$ denotes the vector of differences between two adjacent unknowns defined in the same point on the interface. In the case of perfect bond, the vector $\boldsymbol{c}$ is the zero vector. Substitution of nodal unknowns (8) into the interface condition (13) results in the system

$$\begin{pmatrix} \boldsymbol{B}\boldsymbol{K}^{+}\boldsymbol{B}^{T} & -\boldsymbol{B}\boldsymbol{R} \\ -\boldsymbol{R}^{T}\boldsymbol{B}^{T} & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\alpha} \end{pmatrix} = \begin{pmatrix} \boldsymbol{B}\boldsymbol{K}^{+}\boldsymbol{f} - \boldsymbol{c} \\ -\boldsymbol{R}^{T}\boldsymbol{f} \end{pmatrix} \tag{14}$$

The imperfect bond is characterised by different displacements across material interface. There are several possibilities of evolution of shear stress. The vector of interface slips $\boldsymbol{c}$ can be defined in the form

$$\boldsymbol{c} = \boldsymbol{H}\boldsymbol{\lambda} \tag{15}$$

where $\boldsymbol{H}$ denotes the compliance matrix. Generally, the matrix $\boldsymbol{H}$ can depend on attained Lagrange multipliers $\boldsymbol{\lambda}$. Substitution of (15) to the system (14) results in

$$\begin{pmatrix} \boldsymbol{B}\boldsymbol{K}^{+}\boldsymbol{B}^{T} + \boldsymbol{H} & -\boldsymbol{B}\boldsymbol{R} \\ -\boldsymbol{R}^{T}\boldsymbol{B}^{T} & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\alpha} \end{pmatrix} = \begin{pmatrix} \boldsymbol{B}\boldsymbol{K}^{+}\boldsymbol{f} \\ -\boldsymbol{R}^{T}\boldsymbol{f} \end{pmatrix} \tag{16}$$

If the perfect bond is taken into account, the compliance matrix $\boldsymbol{H}$ is the zero matrix and the classical FETI method is obtained. If the linear law (15) is assumed, stress-slip law is modelled.

In the case of softening branch, the previous definition of the compliance matrix does not work. Sequence of several steps is needed in order to track the softening part of bond slip law. In such case, a proportional load is applied on the structure and its response is computed. Locations with the maximum stresses are determined and factor needed for attainment of the bond stress is evaluated. The applied load is multiplied by this factor. There are some Lagrange multipliers on interface which have the limit magnitude and slip or crack start to grow. Those multipliers are marked and they are removed from the localisation matrices $\boldsymbol{B}$ defined by equation (3). The structure is loaded once again but the matrix $\boldsymbol{B}$ is modified. Locations with the largest stresses are determined and new factor needed for the limit state is evaluated. The applied load is multiplied by this new factor and new point on bond slip curve is obtained. The described algorithm is repeated several times and one or more multipliers are removed in each step.

4 Numerical experiments

In order to check behaviour of the proposed numerical framework, a pull out test with mild hardening is considered. Interaction between concrete and steel reinforcement is assumed. Left figure 1 shows detail of concrete-steel interaction in the case of a slip developed. Finally, the right figure 1 shows stress distribution in pull out test.

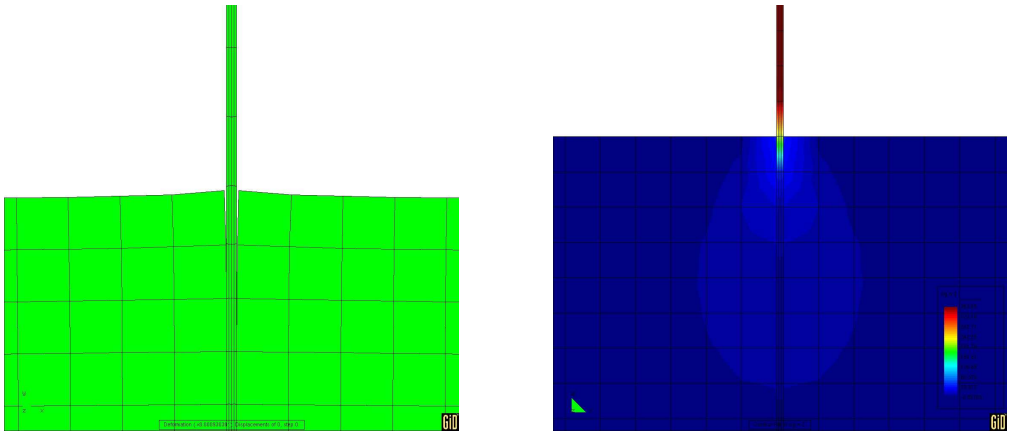


Figure 1: Detail of deformed shape and stress distribution.

5 Conclusions

A numerical framework for description of various bond slip laws between composite matrix and fibres was introduced and implemented. The framework is based on the FETI domain decomposition method which is slightly modified. Compliance matrix or vector of attained interface slips are added to the coarse system of equations. Numerical experiments show better convergence properties of the modified conjugate gradients in many cases. The bond slip laws with hardening can be efficiently described by added compliance. On the other hand, the bond slip laws with softening are modelled by a sequence of steps with reduced stiffness. The proposed framework should be studied in the future because the mesh dependency has to be dealt properly. Similar approaches to the damage or fracture mechanics have to be applied.

Acknowledgement: Financial support for this work was provided by project number P105/11/1160 of Czech Science Foundation. The financial support is gratefully acknowledged.

References

[1] J. Kruis: *Domain decomposition methods on parallel computers.* In: B. H. V. Topping, C. A. Mota Soares (ed.): Progress in Engineering Computational Technology, Saxe-Coburg Publications, Stirlingshire, Scotland, UK, 2004, pp. 299–321.

[2] J. Kruis: *The FETI method and its applications: a review.* In: B. H. V. Topping, P. Iványi (ed.): Parallel, Distributed and Grid Computing for Engineering, Saxe-Coburg Publications, Stirlingshire, Scotland, UK, 2009, pp. 199–216.

[3] J. Kruis, Z. Bittnar: *Reinforcement-matrix interaction modelled by FETI method.* In: U. Langer, M. Discacciati, D. Keyes, O. Widlund, W. Zulehner (ed.): Domain Decomposition Methods in Science and Engineering XVII, Series Lecture Notes in Computational Science and Engineering, Springer-Verlag, Berlin, Germany, 2007, pp. 567–573.

Convergence of finite element methods for nonlinear convective problems

V. Kučera

Faculty of Mathematics and Physics, Charles University in Prague

Abstract

In this short note, we give an overview of the tools needed to estimate the error of finite element methods applied to nonlinear convective problems with smooth solutions. These results along with their generalizations to fully discrete explicit and implicit schemes represent a new, promising technique first outlined by [5] and extended in [4].

1 Continuous problem and discretization

Let $\Omega \subset \mathbb{R}^d, d \in \mathbb{N}$, be a bounded open polyhedral domain. We treat the following nonlinear convective problem. Find $u : \Omega \times (0,T) \to \mathbb{R}$ such that

$$\text{a)} \quad \frac{\partial u}{\partial t} + \operatorname{div} \mathbf{f}(u) = g \quad \text{in } Q_T, \tag{1}$$

$$\text{b)} \quad u\big|_{\Gamma_D \times (0,T)} = 0, \tag{2}$$

$$\text{d)} \quad u(x,0) = u^0(x), \quad x \in \Omega. \tag{3}$$

Here $g : Q_T \to \mathbb{R}$ and $u^0 : \Omega \to \mathbb{R}$ are given functions and $\Gamma_D \subset \partial\Omega$ has positive measure. We assume that the *convective fluxes* $\mathbf{f} = (f_1, \cdots, f_d) \in (C_b^2(\mathbb{R}))^d = (C^2(\mathbb{R}) \cap W^{2,\infty}(\mathbb{R}))^d$, hence $\mathbf{f}$ and $\mathbf{f}' = (f_1', \cdots, f_d')$ are *globally Lipschitz continuous*. The technique presented in [4] allows to generalize the results also to $\mathbf{f} = (f_1, \cdots, f_d) \in (C^2(\mathbb{R}))^d$, i.e. the locally Lipschitz case.

As for the boundary condition (2), we assume in our analysis that $\Gamma_N := \partial\Omega \setminus \Gamma_D$ is an outflow boundary for the exact or approximate solution, i.e. e.g. $\Gamma_N \subseteq \{x \in \partial\Omega;\, \mathbf{f}'(u(x,t)).\mathbf{n} \geq 0\}$.

We discretize problem (1)-(3) using the standard conforming p-order finite element method. Over a quasi-uniform, shape regular, conforming system of triangulations $\{\mathcal{T}_h\}_{h \in (0,h_0)}$, $h_0 > 0$ of $\overline{\Omega}$ we define the space of globally continuous piecewise p-order polynomial functions $S_h = \{v \in C(\overline{\Omega});\, v|_{\Gamma_D} = 0,\, v|_K \in P^p(K) \forall K \in \mathcal{T}_h\}$. We set $h = \max_{K \in \mathcal{T}_h} \operatorname{diam}(K)$. In this function space we introduce the space semidiscrete version of problem (1). We seek $u_h \in C^1([0,T]; S_h)$ such that $u_h(0) = u_h^0 \approx u^0$ and

$$\frac{d}{dt}\big(u_h(t), \varphi_h\big) + b\big(u_h(t), \varphi_h\big) = l\big(\varphi_h\big)(t), \quad \forall \varphi_h \in S_h,\ t \in (0,T). \tag{4}$$

Here, we have introduced the *convective* and *right-hand side forms* defined for $v, \varphi \in H^1(\Omega)$:

$$b(v,\varphi) = -\int_\Omega \mathbf{f}(v) \cdot \nabla\varphi \,\mathrm{d}x + \int_{\Gamma_N} \mathbf{f}(v) \cdot \mathbf{n}\, \varphi \,\mathrm{d}S, \qquad l(\varphi)(t) = \int_\Omega g(t)\varphi \,\mathrm{d}x.$$

We note that a sufficiently regular exact solution u of problem (1) also satisfies (4) for all $\varphi_h \in S_h$, i.e. we have *Galerkin orthogonality property* of the error.

2 Key estimates of the convective terms

As usual in apriori error analysis, we assume that the weak solution u is sufficiently regular:

$$u, u_t \in L^2\big(0,T;H^{p+1}(\Omega)\big), \quad u \in L^\infty(0,T;W^{1,\infty}(\Omega)).$$

Let $\eta_h(t) = u(t) - \Pi_h u(t) \in H^{p+1}(\Omega)$ and $\xi_h(t) = \Pi_h u(t) - u_h(t) \in S_h$, where $\Pi_h v$ is the $L^2(\Omega)$-projection of v on S_h. Then we can write the error e_h as $e_h(t) := u(t) - u_h(t) = \eta_h(t) + \xi_h(t)$. By C we will denote a generic constant independent of h. In our analysis, we shall need the following standard inverse inequalities

$$|v_h|_{H^1} \le C_I h^{-1} ||v_h||,$$
$$\|v_h\|_\infty \le C_I h^{-d/2} \|v_h\|$$

and approximation properties of η, (cf. [2]):

$$\|\eta_h(t)\| \le C h^{p+1} |u(t)|_{H^{p+1}},$$
$$\big\|\tfrac{\partial \eta_h(t)}{\partial t}\big\| \le C h^{p+1} \big|\tfrac{\partial u(t)}{\partial t}\big|_{H^{p+1}},$$

The key estimate of the convective terms is inspired by the work [5], originally derived for the DG method. A complete proof of our case can be found in [4].

Lemma 2.1. *There exists a constant $C \ge 0$ independent of h, t, such that*

$$b\big(u_h(t),\xi(t)\big) - b\big(u(t),\xi(t)\big) \le C\Big(1 + \frac{\|e_h(t)\|_\infty}{h}\Big)\big(h^{2p+1}|u(t)|^2_{H^{p+1}} + \|\xi(t)\|^2\big). \tag{5}$$

Proof. The key trick of the estimate is performing a Taylor expansion of $\mathbf{f}$ with respect to u:

$$\mathbf{f}(u) - \mathbf{f}(u_h) = \mathbf{f}'(u)\xi + \mathbf{f}'(u)\eta - \frac{1}{2}\mathbf{f}''_{u,u_h} e_h^2,$$

where $\mathbf{f}''_{u,u_h}$ is the Lagrange form of the remainder of the Taylor expansion. Substituting into the definition of $b(\cdot,\cdot)$, we obtain the interior terms

$$\int_\Omega \mathbf{f}'(u)\xi\cdot\nabla\xi \,\mathrm{d}x + \int_\Omega \mathbf{f}'(u)\eta\cdot\nabla\xi \,\mathrm{d}x - \frac{1}{2}\int_\Omega \mathbf{f}''_{u,u_h} e_h^2 \cdot\nabla\xi \,\mathrm{d}x.$$

Estimating these terms by (5) is straightforward, using the inverse inequalities and estimates of η. A similar procedure is done for the boundary terms of $b(\cdot,\cdot)$. □

3 Error analysis of the semidiscrete scheme

We proceed similarly as for a parabolic equation. By Galerkin orthogonality, we subtract the equations for u and u_h and set $\varphi_h := \xi_h(t) \in S_h$. Since $\big(\frac{\partial \xi_h}{\partial t}, \xi_h\big) = \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\|\xi_h\|^2$, we get

$$\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\|\xi_h(t)\|^2 = b\big(u_h(t),\xi_h(t)\big) - b\big(u(t),\xi_h(t)\big) - \Big(\frac{\partial \eta_h(t)}{\partial t}, \xi_h(t)\Big).$$

For the last right-hand side term, we use the Cauchy and Young's inequalities and estimates of η and Lemma 2.1 for the convective terms. We integrate from 0 to $t \in [0,T]$,

$$\|\xi_h(t)\|^2 \le C\int_0^t \Big(1 + \tfrac{\|e_h(\vartheta)\|_\infty}{h}\Big)\Big(h^{2p+1}|u(\vartheta)|^2_{H^{p+1}} + h^{2p+2}|u_t(\vartheta)|^2_{H^{p+1}} + \|\xi_h(\vartheta)\|^2\Big)\mathrm{d}\vartheta, \tag{6}$$

where $C \geq 0$ is independent of h, t. For simplicity, we have assumed that $\xi_h(0) = 0$, i.e. $u_h^0 = \Pi_h u^0$. Otherwise we must assume e.g. $\|\xi_h(0)\|^2 \leq Ch^{2p+1}|u^0|^2_{H^{p+1}}$ and include this term in the estimate.

We notice that if we knew *apriori* that $\|e_h\|_\infty = O(h)$ then the unpleasant term $h^{-1}\|e_h\|_\infty$ in (6) would be $O(1)$. Thus we could simply apply the standard Gronwall lemma to obtain the desired error estimates. We state this formally:

Lemma 3.1. *Let $t \in [0, T]$ and $p \geq d/2$. If $\|e_h(\vartheta)\| \leq h^{1+d/2}$ for all $\vartheta \in [0, t]$, then there exists a constant C_T independent of h, t such that*

$$\max_{\vartheta\in[0,t]} \|e_h(\vartheta)\|^2 \leq C_T^2 h^{2p+1}. \tag{7}$$

Proof. The assumptions imply, by the inverse inequality and estimates of η, that

$$\begin{aligned}\|e_h(\vartheta)\|_\infty &\leq \|\eta_h(\vartheta)\|_\infty + \|\xi_h(\vartheta)\|_\infty \leq Ch|u(t)|_{W^{1,\infty}} + C_I h^{-d/2}\|\xi_h(\vartheta)\| \\ &\leq Ch + C_I h^{-d/2}\|e_h(\vartheta)\| + C_I h^{-d/2}\|\eta_h(\vartheta)\| \leq Ch + Ch^{p+1-d/2}|u(\vartheta)|_{H^{p+1}(\Omega)} \leq Ch,\end{aligned}$$

where the constant C is independent of h, ϑ, t. Using this estimate in (6) gives us

$$\|\xi_h(t)\|^2 \leq \tilde{C}h^{2p+1} + C\int_0^t \|\xi_h(\vartheta)\|^2 \,\mathrm{d}\vartheta, \tag{8}$$

where the constants $\widetilde{C}, C$ are independent of h, t. Gronwall's inequality applied to (8) states that there exists a constant $\widetilde{C}_T$, independent of h, t, such that

$$\max_{\vartheta\in[0,t]} \|\xi_h(\vartheta)\|^2 \leq \widetilde{C}_T h^{2p+1},$$

which along with similar estimates for η gives us (7). □

Now it remains to get rid of the *apriori* assumption $\|e_h\|_\infty = O(h)$. For an explicit scheme, this can be done using mathematical induction. Starting from $\|e_h^0\| = O(h^{p+1/2})$, we prove:

$$\|e_h^n\| = O(h^{p+1/2}) \quad \Longrightarrow \quad \|e_h^{n+1}\|_\infty = O(h) \quad \Longrightarrow \quad \|e_h^{n+1}\| = O(h^{p+1/2}).$$

For the method of lines we have continuous time and hence cannot use mathematical induction straightforwardly. However, we can use some continuous version of mathematical induction, cf. [1, 3]. In our case, we can use the simplest version:

Lemma 3.2 (Continuous mathematical induction)**.** *Let $\varphi(t)$ be a propositional function depending on $t \in [0, T]$ such that*

(i) *$\varphi(0)$ is true,*
(ii) *$\exists \delta_0 > 0 :$ $\varphi(t)$ implies $\varphi(t + \delta)$, $\forall t \in [0, T]$ $\forall \delta \in [0, \delta_0] : t + \delta \in [0, T]$.*

Then $\varphi(t)$ holds for all $t \in [0, T]$.

Theorem 3.1 (Semidiscrete error estimate)**.** *Let $p > (1 + d)/2$. Let $h_1 > 0$ be such that $C_T h_1^{p+1/2} = \frac{1}{2} h_1^{1+d/2}$, where C_T is the constant from Lemma 3.1. Then for all $h \in (0, h_1]$ we have the estimate*

$$\max_{\vartheta\in[0,T]} \|e_h(\vartheta)\|^2 \leq C_T^2 h^{2p+1}.$$

Proof. Since $p > (1+d)/2$, h_1 is uniquely determined and $C_T h^{p+1/2} \le \frac{1}{2} h^{1+d/2}$ for all $h \in (0, h_1]$. We define the propositional function φ by

$$\varphi(t) \equiv \Big\{ \max_{\vartheta \in [0,t]} \|e_h(\vartheta)\|^2 \le C_T^2 h^{2p+1} \Big\}.$$

We shall use Lemma 3.2 to show that φ holds on $[0, T]$, hence $\varphi(T)$ holds.

(i) $\varphi(0)$ holds, since this is the error of the initial condition.

(ii) *Induction step*: We fix an arbitrary $h \in (0, h_1]$. Due to the regularity assumptions, the functions $u(\cdot), u_h(\cdot)$ are *uniformly continuous* function from $[0, T]$ to $L^2(\Omega)$. Therefore, there exists $\delta_0 > 0$, such that if $t \in [0, T), \delta \in [0, \delta_0]$, then $\|e_h(t+\delta) - e_h(t)\| \le \frac{1}{2} h^{1+d/2}$. Now let $t \in [0, T)$ and assume $\varphi(t)$ holds. Then $\varphi(t)$ implies $\|e_h(t)\| \le C_T h^{p+1/2} \le \frac{1}{2} h^{1+d/2}$. Let $\delta \in [0, \delta_0]$, then by uniform continuity

$$\|e_h(t+\delta)\| \le \|e_h(t)\| + \|e_h(t+\delta) - e_h(t)\| \le \tfrac{1}{2} h^{1+d/2} + \tfrac{1}{2} h^{1+d/2} = h^{1+d/2}.$$

This and $\varphi(t)$ implies that $\|e_h(s)\| \le h^{1+d/2}$ for $s \in [0, t] \cup [t, t+\delta] = [0, t+\delta]$. By Lemma 3.1, φ holds on $[0, t+\delta]$. This proves the "induction step" $\varphi(t) \Longrightarrow \varphi(t+\delta)$ for all $\delta \in [0, \delta_0]$. □

4 Conclusion

We gave a simple overview of the concepts used to obtain error estimates of smooth solutions of nonlinear convective problems. The results can be extended much further beyond this expository account. For example, for a fully discrete implicit scheme, similar estimates can be obtained after introducing a suitable continuation of the discrete solution. As mentioned, the technique can be extended to locally Lipschitz continuous nonlinearities as well. We refer to [4] for details.

Acknowledgement: The work was supported by the project P201/11/P414 of the Czech Science Foundation. The author is a junior researcher in the University Center for Mathematical Modelling, Applied Analysis and Computational Mathematics (Math MAC).

References

[1] Y. R. Chao: *A note on "Continuous mathematical induction"*. Bull. Amer. Math. Soc. 26 (1), 1919, pp. 17–18.

[2] P. G. Ciarlet: *The finite element method for elliptic problems*. North-Holland, Amsterdam, 1979.

[3] P.L. Clark: *Real induction*.
`http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.187.3514`

[4] V. Kučera: *Finite element error estimates for nonlinear convective problems*. The Preprint Series of the School of Mathematics, preprint No. MATH-knm-2012/1.
`http://www.karlin.mff.cuni.cz/ms-preprints/prep.php`
Submitted to Numer. Math, 2012.

[5] Q. Zhang, C.-W. Shu: *Error estimates to smooth solutions of Runge–Kutta discontinuous Galerkin methods for scalar conservation laws*. SIAM J. Numer. Anal. 42 (2), 2004, pp. 641–666.

Inverse analysis for estimating some characteristics of stress fields

J. Malík, A. Kolcun

Institute of Geonics AS CR, Ostrava

1 Introduction

This work was inspired by the situation which happened during the extraction of the shaft of the Frenštát coal mine in the Beskydy Mountains. During the extraction, when the tube of the shaft ran trough certain geological layers, significant deformations of the concrete lining occurred. The geological structure of the layers mentioned above was complicated and some problems connected with the expected anisotropy of horizontal stress fields were predicted.

Let us start with the description of the technology applied for the shaft. The mobile steel formwork method was applied and the wall of the shaft was reinforced by the concrete lining. The principle of the technology is obvious from Figure 1.

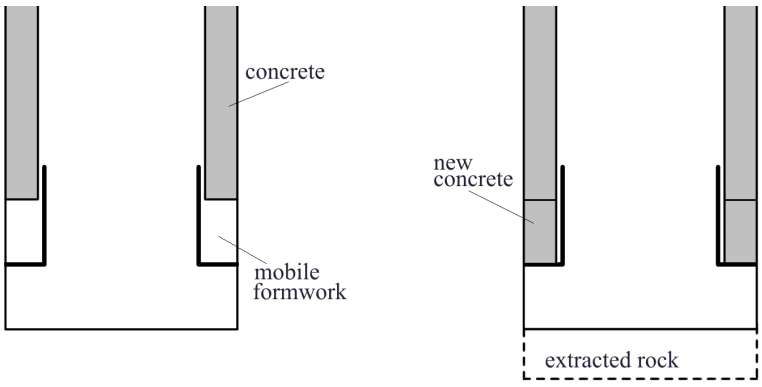


Figure 1: Two stages of technology of reinforcement of the wall of the shaft.

The rock was gradually extracted, the mobile steel formwork was shifted and the vacated room was filled by concrete. After hardening the concrete, the process was repeated.

Because of the original stress in the rock mass some part of the stress present in the rock transfers on the concrete lining, which results in the deformations of the lining. The value of the transferred loads depends on the thickness of extracted rock layer and the application of concrete lining.

2 Basic hypotheses

Let us formulate the basic hypotheses which we will hold in the following mathematical models.

1. The concrete lining is the regular circular ring whose behavior is elastic and material properties are known.

2. The original stress field round the shaft in a certain depth is homogeneous.

3. The loads transferred on the concrete lining result in deformations of the ring and can be approximated by the reduced tensor corresponding to the initial stress tensor. This tensor is a multiplication by of the reduced tensor.

4. The problem is considered to be two dimensional and is analyzed as a 2D elastic problem in every cross section.

The hypothesis 3 yields that we cannot establish the whole stress tensor, but the deformations of the ring give the directions of principal stresses and their ratio.

3 Basic concept of the inverse analysis

As we mentioned in the previous section, our inverse analysis is based on a solution of the 2D elastic problem depicted in Figure 2a).

Let us consider we have material constants of both the concrete and the rock. The square boundary in Figure 2a) is loaded. The loads are derived from the reduced initial stress tensor, so we are going to deal with the first boundary problem. Solutions to the first boundary problem in displacements are not unique and are given up to rigid body translations and rotations (see [1]). Thus we cannot directly use the displacements on the lining, but the changes of possessions between points on the wall of the ring as it is depicted in Figure 2a). We have to transform the measurements into the required form. The date are represented by the matrix $D_{n,5}$, where n it is the number of measurements. The matrix lines are

$$(x_{1,i}, y_{1,i}, x_{2,i}, y_{2,i}, d_i),$$

i.e. coordinates of the pair of the points connected with the $i-th$ measurement and the change of the distance between this pair of points after the deformation of the circle ring. Let us denote the reduced stress tensor

$$\begin{pmatrix} a & c \\ c & b \end{pmatrix} = a \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + b \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} + c \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{1}$$

Our task is to establish the values a, b, c from the analysis of measurements. Our problem is connected with the three 2D boundary value problems, where the loads on the square boundary in Figure 3b) are generated by the three following stress tensors

$$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{2}$$

The solutions u^1, u^2, u^3 of these three problems are unique up to rigid body translations and rotations. The distances between the pairs of the points in which the measurements occurred are independent of rigid body translations and rotations.

The solution u to the general problem, where the loads are generated by the general stress tensor

$$\sigma = \begin{pmatrix} a & c \\ c & b \end{pmatrix}, \tag{3}$$

are then

$$u = au^1 + bu^2 + cu^3.$$

Now we have to choose the parameters a, b, c such that the calculated distances between the pairs of the points, where the measurements occurred, are as near to the measurements as possible.

Thus the three solutions u^1, u^2, u^3 are connected with the three systems $h_i^1, h_i^2, h_i^3, \quad i = 1, \ldots, n$ representing the distances between the pair of the points.

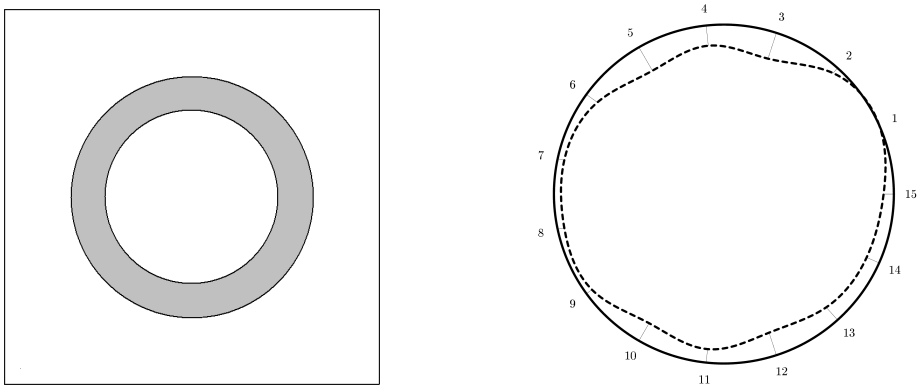
We can choose the parameters a, b, c so that the term

$$\sum_{i=1}^{n} (d_i + ah_i^1 + bh_i^2 + ch_i^3)^2$$

achieves its minimum. We can apply the least square method, which leads to the system of three linear equations and is easy to calculate. Let us notice that the parameters a, b, c do not represent the horizontal stress tensor but the reduced horizontal stress tensor. Because we do not know the magnitude of the loads transferred from the mass to the lining, the parameters a, b, c themselves do not have physical meaning. Nevertheless from the parameters a, b, c we can derive the directions of the principal stresses and the ratio between the principal stresses τ_{max}, τ_{min} which has physical meaning.

We have to consider the thickness of the circle ring is not constant and the application of the least square method eliminates the errors given by the non constant thickness as well as the inaccuracy of measurements.

The method described above was implemented as an additional module in the FEM-code GEM22 which was developed in Institute of Geonics for solving geomechanical problems.



a) b)

Figure 2: Two stages of technology of reinforcement of the wall of the shaft.

This code was applied for the analysis of stress fields in a few cross sections for which the measurements were available (see [2]). The graph depicted in Figure 2b) represents the displacements on the cross section of the shaft in the depth 300 m.

The surrounding rocks and concrete were modeled as isotropic materials whose constants are as follows:

Table 1: Material properties

	E [MPa]	ν
concreete	10000	0.2
rock	10000	0.24

The diameter of the shaft is 8.5 m and the thickness of the lining is 70 cm. The analysis of the situation depicted in Figure 2b) resulted in the ratio between the principal stresses

$$\tau_{max}/\tau_{min} = 3.6,$$

which indicates the considerable anisotropy of the initial stress field in the corresponding geological layer.

4 Conclusion

In this case mathematical modeling is an effective method that makes possible to propose optimal installations of bolt reinforcements. On the other hand the installation above depends very much on the exact knowledge of the principal stress directions which can be detected by the inverse analysis described above. Thus this inverse analysis has to be an essential part of the technology studied in this paper.

The proposed inverse analysis can be applied not only for the situation which we study in this paper, but for tunnels as well. In this case we can analyze the stress fields in the cross sections perpendicular to the tunnel. For such cases we have a reliable estimate of the vertical part of the initial stress, so we can reconstruct the whole initial stress tensor if we have the principal directions and the anisotropy ratio. These values are important for effective installations of rock bolt systems (see [3], [4]).

Acknowledgement: This work has been supported by the grant FR-TI3/579.

References

[1] J. Nečas, I. Hlaváček: *Mathematical Theory of Elasto/Plastic Bodies: An Introduction.* Elsevier, 1981.

[2] Research Report 67 025, Study of Geological and Geomechanical State during Excavation of the Shaft Frenštát, VVUU Ostrava.

[3] J. Malík: *Mathematical Modelling of Rock Bolt Systems I.* Appl.Math. 43, 1998, pp. 413–438.

[4] J. Malík: *Mathematical Modelling of Rock Bolt Systems II.* Appl.Math. 45, 2000, pp. 177–203.

Primární metody rozložení oblasti a hraniční prvky

L. Malý, D. Lukáš

Katedra aplikované matematiky, VŠB-Technická univerzita Ostrava

1 Úvod

Ve své práci se zabývám analýzou a aplikací hraničně prvkového přístupu v primárních metodách rozložení oblasti ve dvou dimenzích, využívaných pro řešení eliptických parciálních diferenciálních rovnic s vysokými skoky v materiálových koeficientech. Řešení pomocí konečných prvků analyzovali a předvedli autoři Bramble, Pasciak a Schatz [1] v roce 1986. Také autoři Toselli a Widlund se zabývali podobnou metodou ve své práci o aditivní Schwarzově teorii [3].

Primární metody rozložení oblasti autorů Brambla, Pasciaka a Schatze na rozdíl od populárnějších FETI metod nezvyšují počet neznámých a zachovávají pozitivní definitnost úlohy, tudíž je lze použít jako předpodmínění v metodě sdružených gradientů. Podstatou metody je aproximace Schurova doplňku. V našem případě se jednotlivé podúlohy snažíme nahradit hraničněprvkovým přístupem a zredukovat tak problém pouze na hranici.

2 Primární metody rozložení oblasti

Řešme parciální diferenciální úlohu ve dvou dimenzích

$$\begin{cases} -\mathrm{div}(k(x)\nabla u(x)) &= f & \text{v } \Omega, \\ u(x) &= 0 & \text{na } \partial\Omega, \end{cases}$$

kde Ω je polygonální oblast a $k(x)$ je po částech konstantní.

Oblast Ω rozdělíme do N nepřekrývajících se podoblastí tak, aby respektovaly skoky v materiálových koeficientech $k(x)$. Řešení takovéto úlohy vede na soustavu lineárních rovnic. Odtud dostaneme matici tuhosti A, kterou můžeme vyjádřit v následujícím přeuspořádaném 2×2 blokovém tvaru

$$A = \begin{pmatrix} A_{II} & A_{I\Gamma} \\ A_{\Gamma I} & A_{\Gamma\Gamma} \end{pmatrix},$$

kde A_{II} odpovídá vnitřním lokálním úlohám na jednotlivých podoblastech, blok $A_{\Gamma\Gamma}$ odpovídá úloze na hranicích a zbývající bloky odpovídají jejich vzájemným interakcím. Matice tuhosti A je symetrická pozitivně definitní.

Stejně si přeuspořádáme vektor pravé strany a dále si řešení vyjádříme jako součet partikulárního a homogenního řešení. Dostáváme soustavu

$$\begin{pmatrix} A_{II} & A_{I\Gamma} \\ A_{\Gamma I} & A_{\Gamma\Gamma} \end{pmatrix} \cdot \begin{pmatrix} u_I^P + u_I^H \\ u_\Gamma^H \end{pmatrix} = \begin{pmatrix} b_I \\ b_\Gamma \end{pmatrix}.$$

Řešení této úlohy spočteme ve třech krocích:

1. $A_{II}u_I^P = b_I$, toto odpovídá úloze $\quad \left\{ \begin{array}{rcll} -k_i\Delta u_i^P & = & f & \text{v } \Omega_i, \\ u_i^P & = & 0 & \text{na } \partial\Omega_i, \end{array} \right.$

2. $Su_\Gamma^H = b_\Gamma - A_{\Gamma I}b_I, \qquad S := A_{SS} - A_{\Gamma I}A_{\Gamma\Gamma}^{-1}A_{I\Gamma}$ (Schurův doplněk),

3. $A_{II}u_I^H = -A_{I\Gamma}u_\Gamma^H$, toto odpovídá úloze $\quad \left\{ \begin{array}{rcll} -k_i\Delta u_i^H & = & 0 & \text{v } \Omega_i, \\ u_i^H & = & u_\Gamma^H & \text{na } \partial\Omega_i. \end{array} \right.$

Efektivní řešení vyžaduje nalézt vhodný předpodmiňovač pro matici A. Její inverzi vyjádříme jako

$$A^{-1} = \begin{pmatrix} I & -A_{II}^{-1}A_{IS} \\ 0 & I \end{pmatrix} \cdot \begin{pmatrix} A_{II}^{-1} & 0 \\ 0 & S^{-1} \end{pmatrix} \cdot \begin{pmatrix} I & -A_{II}^{-1}A_{IS} \\ 0 & I \end{pmatrix}^T.$$

Další postup spočívá v sestavení aproximace Schurova doplňku S. V našem případě je skeleton rozdělen na hrany (Edge - E) a vrcholy (Vertex - V),

$$S = \begin{pmatrix} S_{EE} & S_{EV} \\ S_{VE} & S_{VV} \end{pmatrix} = \begin{pmatrix} I & 0 \\ -R_E & I \end{pmatrix} \cdot \begin{pmatrix} S_{EE} & \tilde{S}_{EV} \\ \tilde{S}_{VE} & \tilde{S}_{VV} \end{pmatrix} \cdot \begin{pmatrix} I & -R_E^T \\ 0 & I \end{pmatrix},$$

kde R_E je lineární interpolace z vrcholu na okolní hrany. Aproximovaný Schurův doplněk

$$\hat{S} := \begin{pmatrix} I & 0 \\ -R_E & I \end{pmatrix} \cdot \begin{pmatrix} \overline{S}_{EE} & 0 \\ 0 & \tilde{S}_{VV} \end{pmatrix} \cdot \begin{pmatrix} I & -R_E^T \\ 0 & I \end{pmatrix}$$

sestává ze dvou částí. $\overline{S}_{EE}$ je bloková diagonální matice lokálních úloh s nulovou Dirichletovou okrajovou podmínkou pro oblasti nad každou hranou. $\tilde{S}_{VV}$ je matice globální Dirichletovy úlohy (hrubého problému) na kostře.

Pro tuto konstrukci předpodmiňovače ve 2d Bramble, Pasciak a Schatz [1] dokázali, že číslo podmíněnosti matice tuhosti je $O((1 + \log(H/h))^2)$, kde H je diametr podoblastí a h je dělení použito při diskretizaci MKP.

3 Numerické výsledky

Výše popsanou metodu jsme otestovali na modelové úloze

$$\left\{ \begin{array}{rcll} -\text{div}(k(x)\nabla u(x)) & = & 1 & \text{v } \Omega, \\ u(x) & = & 0 & \text{na } \partial\Omega, \end{array} \right.$$

kde $\Omega := (a, b) \times (a, b)$ je čtverec rozdělený na $N \times N$ podoblastí (menších stejných čtverců) a $k(x)$ je šachovnicová funkce, která nabývá střídavě hodnot 1 a 1000.

V předpodmiňovači popsaném výše jsme s úspěchem nahradili řešení globální úlohy na skeletonu (hrubého problému) metodou hraničních prvků, tedy matici $\tilde{S}_{VV}$ jsme sestavili pomocí BEM a sledovali počet iterací při řešení celé soustavy. Výsledky jsou obsaženy v Tabulce 1, N je počet dělení čtverce v jednom směru, celkový počet podoblastí je tedy N^2.

$\frac{H}{h}$ \ N	2	3	4	8	16	32
4	4	5	12	13	13	13
8	5	6	14	15	17	17
16	5	7	17	21	21	21
32	6	8	19	25	25	–
64	7	9	22	29	29	–

Tabulka 1: Počty iterací u modelové řlohy.

4 Závěr

Podařilo se nám úspěšně nahradit řešení tzv. hrubého problému metodou hraničních prvků a numericky ověřit očekávané chování při řešení modelové úlohy. Nadále se pokusíme metodou hraničních prvků nahradit také obě lokální úlohy, numericky otestovat efektivnost předpodmiňovače a tím tak doložit analytické odhady.

References

[1] J.H. Bramble, J.E. Pasciak, A.H. Schatz: *The construction of preconditioners of elliptic problems by substructuring I.* In: Mathematics of Computation 47 (175), 1986, pp. 103–134.

[2] J.H. Bramble, J.E. Pasciak, A.H. Schatz: *The construction of preconditioners for elliptic problems by substructuring. II.* Mathematics of Computation 49 (179), 1987, pp. 1–16.

[3] A. Toselli, O. Widlund: *Domain decomposition methods – algorithms and theory.* Springer, 2005.

[4] O. Steinbach, S. Rjasanow: *The fast solution of boundary integral equations.* Springer, 2007.

Parallel implementation of fast boundary element method

M. Merta, D. Lukáš

IT4Innovations & DAM, VŠB-Technical University of Ostrava

1 Introduction

Using the boundary element method (BEM) for a solution of engineering problems we can reduce a dimension of a problem from d to $d-1$. This not only significantly reduces the number of unknowns (when compared to the finite element method) but also the time necessary to generate a mesh. On the other hand, because of the non-locality of the kernel function, conventional BEM produces fully populated matrices with $\mathcal{O}(N^2)$ entries, requiring $\mathcal{O}(N^2)$ operations to assemble, and the same amount of operations per matrix-vector multiplication in iterative solvers. Therefore, its usage for real world problems is limited and some method for matrix sparsification has to be employed [4]. In this work we use our parallel implementation of the fast multipole method (FMM) to solve a boundary value problem for Laplacian with Dirichlet boundary condition.

2 Model problem

We consider the boundary value problem

$$\begin{cases} -\Delta u = 0 & \text{in } \Omega, \\ \gamma^0 u = g & \text{on } \partial\Omega. \end{cases}$$

The solution of this problem can be obtained by the BEM, i.e. using the representation formula

$$u = V\gamma^1 u - K\gamma^0 u,$$

where

$$(Vs)(x) := \int_{\partial\Omega} G(x,y)s(y)\,\mathrm{d}s_y, \quad (Ks)(x) := \int_{\partial\Omega} \frac{\partial G(x,y)}{\partial n_y} s(y)\,\mathrm{d}s_y,$$

with $G(x,y) := \frac{1}{4\pi}\frac{1}{\|x-y\|}$ being the fundamental solution of the Laplace equation in 3D.

For the discretization we use the Galerkin method with piece-wise constant basis and testing functions. After the triangulation $T := \cup_{\ell=1}^{N}\tau_\ell$ we obtain the following system of linear equations

$$V_h t = \left(\frac{1}{2}M_h + K_h\right) g.$$

The matrices V_h, K_h of the single layer and double layer potential, respectively, are given by:

$$V_h^{ij} := \langle V\psi_j, \psi_i\rangle_{\partial\Omega}, \quad K_h^{ij} := \langle K\psi_j, \psi_i\rangle_{\partial\Omega},$$

and M_h is the diagonal matrix with entries $m_{ii} = |\tau_i|$. Because of a nonlocality of the kernel function G, the matrices V_h and K_h are fully populated.

3 Fast multipole method

The fast multipole method introduced by Greengard and Rokhlin [2, 3] uses the fact that the kernel G can be expanded by the spherical harmonic functions

$$\frac{1}{\|x-y\|} \approx \sum_{n=0}^{p} \sum_{m=-n}^{n} \overline{S_n^m}(y) R_n^m(x), \tag{1}$$

$$R_n^{\pm m}(x) = \frac{1}{(n+m)!} \frac{\mathrm{d}^m}{\mathrm{d}u^m} P_n(u)|_{u=\hat{x}_3} (\hat{x}_1 \pm \mathrm{i}\hat{x}_2)^m |x|^n,$$

$$S_n^{\pm m}(y) = (n-m)! \frac{\mathrm{d}^m}{\mathrm{d}u^m} P_n(u)|_{u=\hat{y}_3} (\hat{y}_1 \pm \mathrm{i}\hat{y}_2)^m \frac{1}{|y|^{n+1}},$$

$\hat{y}_i = y_i/\|y\|, \|x\| < \|y\|$. This leads to

$$\int_{\tau_j} \int_{\tau_i} \frac{1}{\|x-y\|} \,\mathrm{d}s_x \,\mathrm{d}s_y \approx \sum_{n=0}^{p} \sum_{m=-n}^{n} \int_{\tau_j} R_n^{\pm m}(x) \,\mathrm{d}s_x \int_{\tau_i} \overline{S_n^{\pm m}}(y) \,\mathrm{d}s_y,$$

which significantly reduces the computational complexity because the integrals by x and y are now decoupled. To guarantee the asymptotic convergence rate of the FMM, the order of expansion p should be chosen proportional to $\log_2 N$.

Since the expansion (1) is only valid for $\|x\| < \|y\|$ we use the recursive geometrical bisection to split a computational domain into clusters and to construct a binary cluster tree. The pair of clusters (C_x, C_y) is said to be *admissible* if it satisfies the condition

$$\min\{\mathrm{diam}C_x, \mathrm{diam}C_y\} \leq \eta \mathrm{dist}(C_x, C_y),$$

otherwise it is called *nonadmissible*. If the pair of clusters is admissible we say that the cluster C_y is in the *farfield* of the cluster C_x and vice versa. Otherwise, they are in their mutual *nearfield*. The admissible cluster pairs correspond to the blocks of matrix approximated by means of FMM.

Using the expansion (1) a matrix-vector multiplication $w = At$ can be evaluated effectively by splitting it into a nearfield and farfield part

$$w_i = \sum_{j \in \mathrm{NF}(i)} A_{ij} t_j + \sum_{j \in \mathrm{FF}(i)} \hat{M}_n^m(O, \psi_i) \tilde{L}_n^m(O, FF(i)),$$

where $NF(i)$, $FF(i)$ are the sets of clusters in the nearfield or farfield, respectively, of the cluster containing the element τ_i. $\hat{M}_n^m(O, \psi_i), \tilde{L}_n^m(O, FF(i))$ denote multipole moments and coefficients of a local expansion associated with a given cluster, respectively. Its efficient computation leverages the existing tree structure:

1. *Upward pass* - multipole moments are computed on the finest level of the tree and translated to the higher levels by multipole to multipole (M2M) translations
2. *Downward pass* - coefficients of a local expansion are computed on the highest posssible level by translation of multipole moments (M2L), and translated to the lower levels by local to local translations (L2L)

Since the multipole coefficients depend on the vector t, these tree traversals have to be repeated in each iteration of an iterative solver. For details see [1].

4 Parallel fast BEM

In this section we briefly describe our parallel implementation of the fast BEM leveraging the fast multipole method. There have been many attempts for a parallel implementation of FMM, usually based on the decomposition of a tree into local subtrees and utilizing space-filling curves to design an efficient communication. These algorithms are able to solve extremely large problems on current supercomputers, however with increasing number of processors the communication deteriates the scalability.

Here we propose a method which is communication-free and leads to an optimal parallel computational scalability $O((n \log n)/N)$ and reasonable memory scalability $O((n \log n)/\sqrt{N})$. The underlying mesh is decomposed into N submeshes and the resulting matrix into corresponding $N \times N$ blocks. Each of N processes is assigned one diagonal block (these are typically most time and memory consuming within the fast BEM) and $N-1$ geometrically closely related off-diagonal blocks, thus the total memory consumption for storing the mesh and related structures is minimal. It turns out that the problem can be formulated in terms of the graph theory as a decomposition of undirected complete graphs. The optimal decomposition is known for N such that it holds

$$\frac{N(N-1)}{2N} = \frac{p(p-1)}{2},$$

where $p+1$ is a power of a prime number.

5 Numerical experiments and conclusion

Our parallel implementation of fast solvers for boundary integral equations was tested on the Vuori cluster located at CSC, Finland, and on the HECToR supercomputer at EPCC, UK. The fast mutlipole method was compared with another method for the sparsification of BEM matrices, adaptive cross approximation (ACA). However, using the FMM, the matrix-vector multiplication in the conjugate gradient algorithm is relatively time-consuming, therefore a usage of a preconditioner seems to be necessary for larger systems.

Acknowledgement: This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and by the project SPOMECH - Creating a multidisciplinary R&D team for reliable solution of mechanical problems, reg. no. CZ.1.07/2.3.00/20.0070 within Operational Programme 'Education for competitiveness' funded by Structural Funds of the European Union and state budget of the Czech Republic. The work was also supported by the Czech Ministry of Education under the project MSM6198910027.

References

[1] G. Of: *Fast multipole method and applications.* In: M. Schanz, O. Steinbach: Boundary element analysis: mathematical aspects and applications, Springer, London, 2007, pp. 135–160.

[2] V. Rokhlin: *Rapid solution of integral equations of classical potential theory.* J. Comput. Phys. 60, 1985, pp. 187–207.

[3] L. Greengard, V. Rokhlin: *A fast algorithm for particle simulations.* J. Comput. Phys. 73, 1987, pp. 325–384.

[4] O. Steinbach, S. Rjasanow: *The fast solution of boundary integral equations.* Springer, 2007.

Error analysis of three methods for the parameter estimation problem based on spatio-temporal FRAP measurement

Š. Papáček, C. Matonoha

University of South Bohemia in České Budějovice, Nové Hrady
Institute of Computer Science AS CR, Prague

1 Introduction

FRAP (Fluorescence Recovery After Photobleaching) measurement technique allows detection of diffusivity (diffusion coefficient D) of autofluorescence molecules or fluorescently tagged compounds (e.g. green fluorescence proteins – GFP) in living cells. This method is based on measurement of the change of fluorescence intensity in a region of interest (ROI – an Euclidian 2D domain) in response to an external stimulus, a short period of high-intensity laser pulse provided by the CLSM.[1] Stimulus, the so-called *bleach*, causes irreversible loss in fluorescence in bleached area without any damage in intracellular structures. After the bleach, the observed recovery in fluorescence reflects diffusion of fluorescence compounds from the area outside the bleach. Based on spatio-temporal FRAP images, the diffusion is reconstructed using either a closed form model or simulation based model. In the latter case, beside a single diffusion coefficient D, also the sequence $\{D_j\}$ can be estimated as well. Let us underline that FRAP images are in general very noisy, with small signal to noise ratio (SNR), i.e. in order to get reliable results for the sequence $\{D_j\}$, an adequate technique residing in *regularization* is mandatory.

2 Inverse problem formulation

Assuming (i) local homogeneity, i.e. the concentration profile of fluorescent particles is smooth, (ii) isotropy, i.e. diffusion coefficient D within the domain Ω is space-invariant, (iii) an unrestricted supply of unbleached particles outside of the target region (i.e. assuring the complete recovery), the following dimensionless diffusion equation describes the unbleached particle concentration $y(r,t)$: $\frac{\partial y}{\partial t} - \nabla \cdot (D \nabla y) = 0$. Moreover, for all three further studied methods, we assume the special geometry residing in one-dimensional simplification getting y as a function of dimensionless quantities: spatial coordinate x, time τ and re-scaled diffusion coefficient p:

$$\frac{\partial y}{\partial \tau} - p \frac{\partial^2 y}{\partial x^2} = 0 \ , \tag{1}$$

where $x := \frac{r}{L}$, L is a characteristic length, $\tau := t/T$, T is a constant with some characteristic value (e.g. time interval between two measurements), and $p := D\frac{T}{L^2}$.

The initial condition and Dirichlet boundary conditions are:

$$y(x, \tau_0) = f(x), \quad x \in [0,1], \tag{2}$$

$$y(0,\tau) = g_0(\tau), \quad y(1,\tau) = g_1(\tau), \quad \tau \geq \tau_0. \tag{3}$$

[1] Confocal laser scanning microscopy (CLSM) allows the selection of a thin cross-section of the sample by rejecting the information coming from the out-of-focus planes. However, the small energy level emitted by the fluorophore and the amplification performed by the photon detector introduces a measurement noise.

Parameter estimation, ill-posedness and error analysis

The inverse problem studied is to estimate the model parameter p^* (generally a vector) from time course of the signal $y(x,\tau)^\delta$ observed at various time instants. The available data $y(x,\tau)^\delta$ are noisy and δ plays the role of a bound on the data noise (later on also the error variance σ_0 will be introduced). Some methods based on FRAP data do not use all measured values of $y(x,\tau)^\delta$, hence we further define the observation operator G that evaluates y on certain space-points $i \in \{1,\dots,n\}$ and time-points $j \in \{1,\dots,m\}$ where the experimental observations (also referred to as the model output) are taken, i.e. $G(y_{i,j}) = z(\tau_j)$. Denoting by $p = (p_1,\dots,p_q)$ the parameter vector, the inverse problem can be formulated as a system of non-linear equations:

$$F(p) = z^\delta, \quad F = G \circ S. \tag{4}$$

Here, $F = G \circ S$ represents the parameter-to-output map, defined as the concatenation of the PDE solution operator S onto the solution vector y of the underlying system (1)-(3), i.e. $S(p) = y_{i,j}$ and the observation operator G. Due to noisy data and model imperfections, the system (4) is replaced by a nonlinear least squares problem where $\| \, . \, \|$ is an appropriate norm for measuring the discrepancy between data and simulated output:

$$\| \, z^\delta - F(p) \, \|^2 \rightarrow \min_{\,p>0} \tag{5}$$

The inverse problem (5) is ill-posed in the sense that its solution (in the least squares sense) does not depend continuously on the data, i.e. noisy data as well as round-off errors may be amplified by an arbitrarily large factor. In order to overcome these instabilities the following regularization method is proposed:

$$\| \, z^\delta - F(p) \, \|^2 + \alpha \, \| \, p - p_0 \, \|^2 \rightarrow \min_{\,p,p_0>0} \tag{6}$$

where the positive regularization parameter α enforces stable dependency of p_α^δ (the solution to (6)) on the noisy data z^δ and p_0 represents an a-priory guess subjected to the minimization.

The above described method of Tikhonov regularization [7] was studied in our paper [5], however the error concepts were not treated there. In the next section we perform the error analysis for three FRAP methods exploiting properties of the sensitivity matrix $\chi = \frac{\partial z}{\partial p}$, i.e., the Jacobian matrix of the output, being evaluated at p_0.[2] More precisely,

$$\chi_{jk}(p_0) = \frac{\partial z(\tau_j;p)}{\partial p_k} \, |_{p=p_0}, \quad 1 \le j \le m, \quad 1 \le k \le q. \tag{7}$$

The statistical model for the observation process is following: $z_j^\delta = z(\tau_j;p_0) + \varepsilon_j$. Moreover, assuming $E[\varepsilon_j] = 0$, $\mathrm{var}(\varepsilon_j) = \sigma_0^2 < \infty$, $\mathrm{cov}(\varepsilon_j,\varepsilon_k) = 0$ whenever $j \neq k$, we have $E[z_j^\delta] = z(\tau_j;p_0)$, $\mathrm{var}(z_j^\delta) = \sigma_0^2$. The solution to (6) obtained using data z^δ is denoted as $\hat{p}$ and is used in the calculation of error variance and $q \times q$ covariance matrix $\Sigma_0 = \mathrm{cov}(p_i,p_j)$, i.e., σ_0^2 is approximated by $\hat{\sigma}^2 = \frac{1}{m-q}|z^\delta - z(\hat{p})|^2$, and Σ_0 is approximated by $\hat{\Sigma}_0 = \hat{\sigma}^2 \left[\chi(\hat{p})^T \chi(\hat{p})\right]^{-1}$. The standard errors of parameters p_k used to quantify uncertainty in the estimation are

$$SE_k(\hat{p}) = \hat{\sigma}\sqrt{[\chi(\hat{p})^T\chi(\hat{p})]^{-1}_{kk}}, \quad 1 \le k \le q. \tag{8}$$

[2] Let see the first order Taylor approximation $\Delta z \approx \chi \Delta p$ relates the perturbation. Accordingly to [1], a parameter vector is defined as *sensitivity identifiable* if $\Delta z \approx \chi \Delta p$ can be solved uniquely (in the local sense) for Δp. Moreover, a sufficient condition for sensitivity identifiability is the nonsingularity of the Fisher information matrix $FIM = \chi^T\chi$ (or equivalently $\det(\chi^T\chi) \neq 0$), i.e., one sees that parameter estimation depends inherently on the condition number of *FIM*.

The propagation of uncertainty from the observation process to the estimated parameter vector is induced by $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_m)^T$ in equation (for more details, see [1, 6]):

$$p \approx p_0 + \left[\chi(p_0)^T \chi(p_0)\right]^{-1} \chi(p_0)^T \varepsilon. \tag{9}$$

3 Three FRAP methods: assessing uncertainty

C. W. Moulineaux *et al.*, Nature (1997)

C. W. Moulineaux *et al.* [4] have measured one-dimensional bleaching profiles (with common variance σ_0^2) along the specimen long axis. Supposing both the infinite domain ($r \in R$) and initial Gaussian bleaching profile, i.e. $y(r, t_0) = y_{0,0} \exp \frac{-2r^2}{{r_0}^2}$, then the solution $y(r, t)$ of diffusion equation (1) is $y(r, t) = \frac{y_{0,0} r_0}{\sqrt{{r_0}^2 + 8Dt}} \exp \frac{-2r^2}{{r_0}^2 + 8Dt}$. The time evolution of maximum depth $y(0, t)$, which was taken as the single observed data point $z(t)$ at time t, and the Fisher information matrix $FIM = \chi^T \chi$ (which collapses to a scalar for $q = 1$), are given by:

$$z_M(t) = \frac{y_{0,0} r_0}{\sqrt{{r_0}^2 + 8Dt}}, \quad FIM_M = \sum_{j=1}^{m} \left[\frac{\partial z_M(t_j)}{\partial D}\right]^2 = \sum_{j=1}^{m} \left[\frac{4 y_{0,0} r_0 t_j}{({r_0}^2 + 8Dt_j)^{3/2}}\right]^2. \tag{10}$$

The weighted linear regression is used in [4] to estimate diffusion coefficient $\hat{D}$ and accordingly to (8) we quantify its standard error: $SE(\hat{D}) = \sigma_0 / \sqrt{FIM_M}$.

J. Ellenberg *et al.*, J. Cell Biol. (1997)

The Ellenberg *et al.* (1997) method [2] to calculate the diffusion coefficient D for stripe ROI is based on the fluorescent signal integrated from the whole ROI (2D domain Ω): $frap(t) = \int_\Omega y(r, t) \mathrm{d}S$ and is normalized as follows: $frap(t_0) = 0$, $frap(\infty) = 1$. Assuming the bleach is complete, there is no immobile fraction, the cell is uniform rectangle, the bleached strip is perpendicular to the long direction, then plot of this so-called *FRAP recovery curve* against time should give a saturation curve according to the formula: $frap(t) = 1 - \sqrt{(w^2/(w^2 + 4\pi Dt))}$, where w is the stripe width. Introducing the dimensionless variables $2L := w$, $p := \frac{D}{D_0}$, $\tau := t \frac{D_0}{L^2}$ we have similarly as in (10), with reduced variance $\sigma_E^2 = \frac{\sigma_0^2}{\sqrt{n}}$ (n is the number of observed data points integrated into $z_E(\tau_j)$ at each time instant τ_j):

$$z_E(\tau) = 1 - \frac{1}{\sqrt{1 + p\pi\tau}}, \quad FIM_E = \sum_{j=1}^{m} \left[\frac{\frac{1}{2}\pi\tau_j}{(1 + p\pi\tau_j)^{3/2}}\right]^2, \quad SE(\hat{p}) = \frac{\sigma_E}{\sqrt{FIM_E}}. \tag{11}$$

FD approximation of PDE (1-3) & Tikhonov regularization based method [5]

As the analytical approach has several limitations (e.g. cell geometry restriction, full recovery is required, bleach profile must be gaussian-like, etc.) we model the process by the Fickian diffusion equation with realistic initial and boundary conditions instead, and the parameter estimation is formulated as an ordinary least squares problem with (6) or without (5) regularization. By this way the sequence of parameters $p_k, 1 \le k \le q$, is determined and the uncertainty assessment based on $q \times q$ Fisher information matrix is led similarly as in above cases, see (7)–(9).

4 Discussion

We have presented three methods for the estimation of diffusivity of fluorescent compounds based on spatio-temporal FRAP measurement and the pertinent error analysis as well. The first two methods, representing the state-of-the-art in FRAP measurement, are based on the curve fitting to an analytical (closed form) models, and obviously need some unrealistic or hard-to-accomplish conditions to be supposed. Our third method is based on finite difference approximation of PDE describing the diffusion process (with the diffusion coefficient D as a parameter) and on the minimization of an objective function evaluating both the disparity between the experimental and simulated time-varying concentration profiles and the smoothness of the time evolution of D as well. This latter approach naturally takes into account both the specimen geometry and time-dependent Dirichlet boundary conditions. The uncertainty assessment is based on the sensitivity matrix calculated either analytically (mainly in case of the curve fitting to an algebraic formula) or numerically.[3] Furthermore, the error analysis provides the tool for discerning among different methods.

Acknowledgement: This work was supported by the project "Jihočeské výzkumné centrum akvakultury a biodiverzity hydrocenóz" (CENAKVA CZ.1.05/2.1.00/01.0024), OP VaVpI and by the long-term strategic development financing of the Institute of Computer Science (RVO:67985807).

References

[1] A. Cintrón-Arias, H. T. Banks, A. Capaldi, A. L. Lloyd: *A sensitivity matrix based methodology for inverse problem formulation.* J. Inv. Ill-Posed Problems 17, 2009, pp. 545–564.

[2] J. Ellenberg, E. D. Siggia, J. E. Moreira, C. L. Smith, J. F. Presley, H. J. Worman, J. Lippincott-Schwartz: *Nuclear membrane dynamics and reassembly in living cells: targeting of an inner nuclear membrane protein in interphase and mitosis.* The Journal of Cell Biology 138, 1997, pp. 1193–1206.

[3] L. Lukšan, M. Tůma, J. Vlček, N. Ramešová, M. Šiška, J. Hartman, C. Matonoha: *UFO 2011 – Interactive system for universal functional optimization.* Technical Report V-1151, Institute of Computer Science, Academy of Sciences of the Czech Republic, Prague 2011 `http://www.cs.cas.cz/luksan/ufo.html`.

[4] C. W. Moullineaux, M. J. Tobin, G. R. Jones: *Mobility of photosynthetic complexes in thylakoid membranes.* Nature 390, 1997, pp. 421–424.

[5] Š. Papáček, R. Kaňa, C. Matonoha: *Estimation of diffusivity of phycobilisomes on thylakoid membrane based on spatio-temporal FRAP images.* Mathematical and Computer Modelling 2012, doi:10.1016/j.mcm.2011.12.029.

[6] G. A. F. Seber, C. J. Wild: *Nonlinear regression.* John Wiley & Sons, Chichester, 2003.

[7] A. N. Tychonoff, V. Y. Arsenin: *Solution of ill-posed problems.* Washington, Winston & Sons, 1977.

[3] Although the semi-analytical method for calculating sensitivities $\frac{\partial z(\tau_j;p)}{\partial p_k} \mid_{p=p_0}$ based on (1)-(3) exists (it resides in the first order ODE for the sensitivities dynamics, see [1]), let us remark that the evaluation of the sensitivity matrix $\chi(p_0)$ or FIM can be done during the optimization procedure (5) for free, because it is done inside the UFO procedure [3].

Problem of identification of heat transfer coefficients

P. Salač

Technical University of Liberec

1 Introduction

This work concerns an identification of the heat coefficients between a cast, a mould, and an environment. The goal of the identification is to find the flux density of modified mass coefficient function of curing melt and the coefficient function of the heat-transfer between the mould and the environment. The aim is to modify these coefficients to achieve the fixed given values at n given points that were obtained as the mean values of measured courses of temperature by sensors.

Mathematical model is a strong idealization of a non-stationary periodical problem of the heat conduction. We study the problem of the stationary conduction of the heat for mean values of this periodical process.

The cost functional is defined as the squared L^2 norm of the difference between a given interpolation function and the calculated temperature.

We define a weak formulation of the state problem and formulate the problem of the identification of the heat-transfer coefficient and the flux density of modified mass of the body coefficient.

2 Formulation of the problem

We assume the problem of steady heat conduction in the union of two regions $\Omega = \Omega_0 \cup \Omega_1$. We assume the existence of a heat source with given density q in the inner region Ω_0 and no heat source in the region Ω_1. We divide the notion for a searched function ϑ, representing distribution of temperature in the system, into the sum of two functions as

$$\vartheta = \vartheta_0 + \vartheta_1 \ ,$$

where

$$\vartheta_i = \begin{cases} \vartheta|_{\Omega_i} & \text{in} \quad \Omega_i \\ 0 & \text{in} \quad \Omega \setminus \Omega_i \end{cases} \qquad \text{for} \quad i = 0,\, 1\,. \tag{1}$$

Further we denote by $\vartheta_i|_{\Gamma_j}$ the trace of the solution ϑ_i on the boundary Γ_j if Γ_j is a boundary of Ω_i for $i,\ j$. We assume the steady heat conduction problem

$$-k_0 \Delta \vartheta_0 = q \qquad \text{in } \Omega_0 \ , \tag{2}$$

$$-k_1 \Delta \vartheta_1 = 0 \qquad \text{in } \Omega_1 \ , \tag{3}$$

where $k_0,\ k_1$ are coefficients of thermal conductivity in the regions $\Omega_0,\ \Omega_1$ and $q \in L^2(\Omega_0)$ is the given function.

The heat-transfer through the boundary Γ_1 (i. e. between the mould and the environment) is modeled as a boundary condition of the third kind of the contact between a body and an

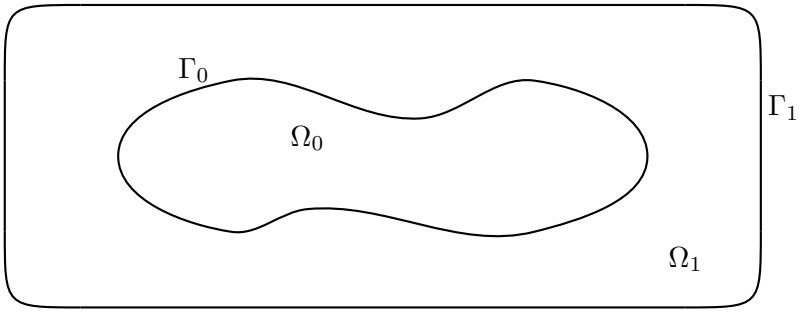


Figure 1: *Scheme of the mould and the cast.*

environment (see [2]), thus

$$-k_1\frac{\partial\vartheta_1}{\partial n} = \alpha(\vartheta_1|_{\Gamma_1} - \vartheta_E) \qquad \text{on} \quad \Gamma_1 \ , \tag{4}$$

where $\frac{\partial}{\partial n}$ denotes the derivative according to the outward normal with respect to the region Ω_1, $\alpha > 0$ denotes the coefficient of the heat-transfer between the mould and the environment, $\vartheta_1|_{\Gamma_1}$ is a trace of ϑ_1 on the boundary Γ_1 of the region Ω_1 and $\vartheta_E > 0$ a temperature of an environment.

We use the transit condition for contact between two bodies, where one of them changes its state of matter because of the influence of solidification (see [2]), to describe transfer of heat through the boundary Γ_0 between the cast and the mould. Thus

$$k_1\frac{\partial\vartheta_1}{\partial n} - k_0\frac{\partial\vartheta_0}{\partial n} \quad = \quad \beta \qquad \text{on} \quad \Gamma_0 \ , \tag{5}$$

where $\beta > 0,\ \beta \in C^{(0),1}(\Gamma_0)$ represents the flux density of modified mass of the body, $\frac{\partial}{\partial n}$ denotes the derivative according to the outward normal with respect to the region Ω_1, resp. Ω_0.

We define the set of admissible functions as

$$\begin{aligned} U_{ad}^{\alpha\beta} = \{\ & (\alpha,\, \beta) \in C^{(0),1}(\Gamma_1) \times C^{(0),1}(\Gamma_0)\,; \\ & \text{(i. e. Lipschitz functions according to the length of relevant boundary)}, \\ & 0 < \alpha_{\min} \le \alpha \le \alpha_{\max},\ |\alpha'| \le C_1 \ ,\ 0 < \beta_{\min} \le \beta \le \beta_{\max},\ |\beta'| \le C_2\} \ , \end{aligned}$$

where the function α represents the heat-transfer coefficient on the boundary Γ_1 and β represents the flux density of modified mass of the cast on the boundary Γ_0.

We define the operators

$$\begin{aligned} \mathrm{En}(\vartheta,\, \psi) = \ & k_0 \int_{\Omega_0} \left(\frac{\partial\vartheta_0}{\partial x}\frac{\partial\psi}{\partial x} + \frac{\partial\vartheta_0}{\partial y}\frac{\partial\psi}{\partial y} + \frac{\partial\vartheta_0}{\partial z}\frac{\partial\psi}{\partial z} \right) d\Omega + \\ & + \ k_1 \int_{\Omega_1} \left(\frac{\partial\vartheta_1}{\partial x}\frac{\partial\psi}{\partial x} + \frac{\partial\vartheta_1}{\partial y}\frac{\partial\psi}{\partial y} + \frac{\partial\vartheta_1}{\partial z}\frac{\partial\psi}{\partial z} \right) d\Omega \ , \end{aligned} \tag{6}$$

$$\mathrm{Ev}(\vartheta,\, \alpha,\, \psi) = \int_{\Gamma_1} \alpha\vartheta_1|_{\Gamma_1}\psi \, dS \ , \tag{7}$$

$$\mathrm{So}(\psi) = \varrho_1 \int_{\Omega_0} q\psi \, d\Omega \ , \tag{8}$$

$$\mathrm{Coef}_\Omega(\alpha,\, \beta,\, \psi) = \int_{\Gamma_1} \alpha\vartheta_E\psi \, dS + \int_{\Gamma_0} \beta\psi \, dS \ , \tag{9}$$

to define the state problem based on the variational formulation of the heat transfer equation.

The State Problem:
We look for the function $\vartheta \equiv \vartheta(\alpha,\, \beta) \in H^1(\Omega)$ such that

$$\mathrm{En}(\vartheta,\, \psi) \;+\; \mathrm{Ev}(\vartheta,\, \alpha,\, \psi) = \mathrm{So}(\psi) + \mathrm{Coef}_\Omega(\alpha,\, \beta,\, \psi) \qquad \forall \psi \in H^1(\Omega)\,, \tag{10}$$

where $(\alpha,\, \beta) \in U_{ad}^{\alpha\beta}$.

Theorem 1. (existence and uniqueness of the solution of the state problem)
The state problem (10) has a unique solution $\vartheta(\alpha,\, \beta)$ for each $(\alpha,\, \beta) \in U_{ad}^{\alpha\beta}$.

Proof. It is sufficient to verify the assumptions of the Lax-Milgram Theorem. □

We assume that the temperatures are given in the set of n points. This data can be obtained from the measurements done in the mould during the production cycle as a mean of values of periodic time dependent functions. We denote temperatures at the given points $t(z_i)$, for $i = 1, \ldots, n$. We assume the existence of a function $\kappa \in C(\Omega)$ such that $\kappa(z_i) = t(z_i)$ for $i = 1, 2, \ldots, n$ (κ can be an interpolation function obtained from the measured values).

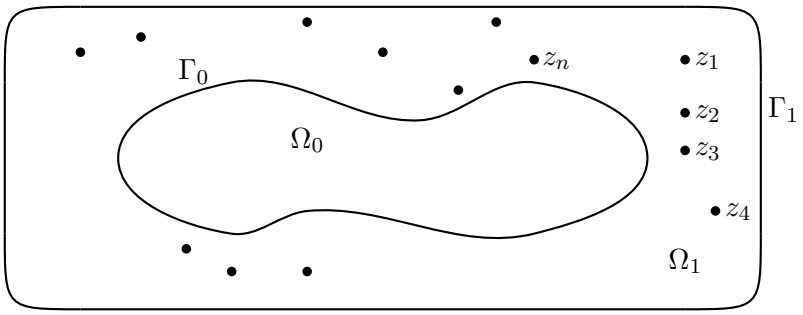


Figure 2: *Scheme of the mould and the cast with the points of measurements.*

We define **the cost functional** as

$$\mathcal{J}^{\mathcal{I}}(\alpha,\, \beta) = \|\vartheta(\alpha,\, \beta) - \kappa\|^2_{1,\Omega}\;, \tag{11}$$

where $\vartheta(\alpha,\, \beta)$ is the solution of the state problem (10).

Now we formulate **the problem of identification :**
We look for the optimal design $(\alpha^*,\, \beta^*) \in U_{ad}^{\alpha\beta}$ such that

$$\mathcal{J}^{\mathcal{I}}(\alpha^*,\, \beta^*) \le \mathcal{J}^{\mathcal{I}}(\alpha,\, \beta) \qquad \forall (\alpha,\, \beta) \in U_{ad}^{\alpha\beta}\;. \tag{12}$$

Theorem 2. (existence of solution of the identification problem)
The problem (12) has at least one solution.

Proof. We use Theorem 2.1. published in [1] page 29. □

3 Conclusion

Presented contribution introduces some theoretical way how to estimate unknown values of the coefficient of the flux density of modified mass coefficient function of curing melt and the coefficient function of the heat-transfer between the mould and the environment. Both the coefficients play an important role in calculating the temperature distribution in models of technological process of casting into moulds. Unfortunately, there is no direct method to measure these coefficients that increases the importance of the described identification problem.

Acknowledgement: The paper was supported by the project ESF, no. CZ.1.07/2.3.00/09.0155, "Constitution and improvement of a team for demanding technical computations on parallel computers at TU Liberec".

References

[1] J. Haslinger, P. Neittaanmäki: *Finite element approximation for optimal shape design: theory and applications.* In: John Wiley & Sons Ltd., Chichester, 1988.

[2] S.N. Šorin: *Sdílení tepla.* In: Nakladatelství technické literatury, Praha, 1968.

Weak solutions for a class of nonlinear integrodifferential equations

I. Soukup

Faculty of Mathematics and Physics, Charles University in Prague

Introduction

Presented work investigates a system of evolutionary nonlinear partial integrodifferential equations in three dimensional space. In particular it studies the existence of a solution to the system introduced in [1] with Dirichlet boundary condition and given initial condition. The studied model represents so-called generalized integral Oldroyd-type model for incompressible viscoelastic nonnewtonian fluids.

The main goal of this work is to give a deeper theoretical knowledge about the properties of mentioned model, which is one of many models describing such fluids like blood (especially in thin veins) or large variety of industrial materials. We feel obligated to emphasize that this work is purely analytical and do not study given model from the perspective of physics or mathematical modelling.

1 Mathematical formulation of the problem

We are looking for a couple $(\mathbf{u}, \pi)$ satisfying the following system of equations

$$\partial_t \mathbf{u} + \mathbf{u}\nabla\mathbf{u} = -\nabla\pi + \operatorname{div}\mathbf{F}(\nabla\mathbf{u}) + \int_0^t G(t-s)\operatorname{div}\mathbf{H}(\nabla\mathbf{u})\,ds + \mathbf{f},$$

$$\operatorname{div}\mathbf{u} = 0,$$

in $(0,T)\times\Omega$, where $\Omega \subset \mathbb{R}^3$ is a bounded domain with a sufficiently smooth boundary. We consider the system of equations with Dirichlet boundary condition

$$\mathbf{u}|_{\partial\Omega} = 0 \qquad \text{in } (0,T)$$

and initial condition

$$\mathbf{u}(0) = \mathbf{u}_0 \qquad \text{in } \Omega\,.$$

The operators F and H are generally nonlinear (usually power-like) and function H is a so-called scalar kernel. Vector functions $\mathbf{f}$ and $\mathbf{u}_0$ are given.

2 Current state of knowledge

This problem was first studied by T. Bárta in [1] and it is so far the first and the last research paper studying given model in the precise form we described above. In particular, T. Bárta succesfully proved the existence of weak solutions under the assumption of p-power-like behaviour of the nonlinearities F and H, i.e. he assumed p-boundedness and p-lipschitz continuity of

nonlinear operators F and H and monotonicity of F, where by the index p we emphasize that the mentioned properties were dependent on some parameter p. The main idea of the proof is based on the ideas of Ladyzhenskaya she used in her studies of similar problem with the integral term missing. She also assumed monotone and p-bounded nonlinearity F (to demonstrate properly the importance of parameter p, we present in what sense we understand mentioned p-boundedness, i.e. F satisfies inequality $\|F(\nabla \mathbf{u})\|_{p'} \leq C\|\nabla \mathbf{u}\|_p^{p-1}$ for appropriate $\mathbf{u}$). She showed the existence of global solution for values of parameter $p \geq \frac{11}{5}$ (see [3], [4], [5]). The results of T. Bárta also holds for the similar values of parameter p because of the same structure of proof.

The Results of Ladyzhenskaya were later extended to values $p \geq 2$ by Málek, Nečas and Růžička (see [6]) under additional assumptions on F and then by Wolf (see [7]) to $p \geq 8/5$ without any additional assumptions. Moreover, just recently was the result of Wolf extended even for $p \geq 6/5$ (see [2]) by Diening, Růžička and Wolf without additional assumptions on F.

Nevertheless, the model in the form we are investigating was studied only by T. Bárta where the existence of a weak solution was proven in three dimensional space with the assumption $p \geq 11/5$ as we already mentioned.

In our work we focus on the result of Málek, Nečas and Růžička and adopting their method we aim to the same results for the integrodifferential model.

Thus, our main goal will be the improvement of Bárta's result using the method from Málek, Nečas and Růžička. In particular we will try to obtain existence of weak solutions for $p \in [2, \frac{11}{5})$ and regularity properties for higher values of the parameter p. As we mentioned, we will proceed along the lines of [6]. The main difference compared to [6] is the presence of the integral term.

3 Scheme of the proof

We adopt the scheme of the proof from [6] and try to avoid the complications rising from the presence of the integral term. The procedure consists of an approximation of the convective term and an approximation of the potentials of nonlinearities F and H using a quadratic function, proving the existence of the approximative solution and then returning to the original problem via regularity of the approximative solution and properties of the nonlinearities.

Acknowledgement: The work was supported by the grant SVV-2012-265316.

References

[1] T. Bárta: *Generalized integral Oldroyd-type models for viscoelastic fluids.* Charles University in Prague, Prague, 2009.

[2] L. Diening, M. Růžička, J. Wolf: *Existence of weak solutions for unsteady motions of generalized Newtonian fluids.* Ann. Sc. Norm. Super. Pisa Cl. Sci. (5) 9, (1) 146, 2010.

[3] O. A. Ladyzhenskaya: *On some new equations describing dynamics of incompressible fluids and on global solvability of boundary value problems to these equations.* Trudy Steklov's Math. Institute 102, 1967, pp. 85–104.

[4] O. A. Ladyzhenskaya: *On some modifications of Navier-Stokes equations for large gradients of velocity.* Zapiski Nukhnych Seminarov LOMI 7, 1968, pp. 126–154.

[5] O. A. Ladyzhenskaya: *The Mathematical Theory of Viscous Incompressible Flow.* Gordon and Breach, New York, 1969.

[6] J. Málek, J. Nečas, M. Růžička: *On weak solutions to a class of non-newtonian incompressible fluids in bounded three-dimensional domains: the case* $p \geq 2$. Advences in Differential Equations 6 (3), 2001, pp. 257–302.

[7] J. Wolf: *Existence of weak solutions to the equations of non-stationary motion of non-Newtonian fluids with shear rate dependent viscosity.* J. Math. Fluid Mech. 9 (1), 2007, pp. 104–138.

Numerical solution of contact perfectly plastic problems: part I – theory and numerical methods

S. Sysala, J. Haslinger, M. Čermák

Institute of Geonics AS CR, Ostrava
Charles University in Prague
IT4Innovations, VŠB-Technical University of Ostrava

1 Introduction

Our contribution is divided into two parts. In Part I, we focus on the theory of discretized problems and suitable numerical methods. In Part II, see [2], we describe implementation of the problem and illustrate it on model examples.

In Section 2, we formulate the primal and dual formulation of the problem. Further we summarize relations among them and their solvability in dependence on the parameter of proportional loading. In Section 3, we consider a finite element discretization of the problem. We extend existence results for the primal formulation (in terms of displacements). We also describe a one-to-one relation between the load parameter and the work of external forces. In Section 4, we introduce a modified semi-smooth Newton method for solving the problem and present convergence results.

2 Formulation of the problem

We consider 3D contact problem for two elastic-perfectly plastic bodies Ω^1, Ω^2 with bounded contact zones Γ_c^1, Γ_c^2, frictionless contact boundary conditions, the Hencky model with the von Mises plastic yield criterion and the small strain assumption. The bodies are fixed on Γ_u^1, Γ_u^2 and subject to external forces which are proportionally increasing from 0 up to the so-called limit load. We investigate the problem in dependence on the loading parameter $\lambda \in [0, \overline{\lambda}]$. For more details, we refer [9, 5] or [6].

To formulate the problem, we introduce the following functional spaces:

$$S = \left\{ \tau = (\tau_{ij}) : \Omega \to \mathbb{R}^{3\times 3}_{sym} \mid \ \tau_{ij}|_{\Omega^k} \in L^2(\Omega^k) \ \ \forall i,j = 1,2,3,\ k = 1,2 \right\}, \quad \Omega = \Omega^1 \cup \Omega^2,$$

representing the stress and strain fields with the scalar product $\langle \tau, e \rangle = \int_\Omega \tau : e \, \mathrm{d}x$ and the norm $\|e\|_E = \sqrt{\langle \mathbb{C}e, e \rangle}$, where $\mathbb{C}$ represents the elasticity tensor for an isotropic material. Further

$$V = \{ v \mid \ v|_{\Omega^k} \in \left(H^1(\Omega^k) \right)^3,\ k = 1,2,\ v = 0 \text{ on } \Gamma_u^1 \cup \Gamma_u^2 \}$$

representing the displacement fields with the energy norm $|||v||| := \|\varepsilon(v)\|_E$, $v \in V$. In V and S, we define the convex sets of kinematically admissible displacement fields, plastically and statically admissible stress fields, respectively:

$$\begin{aligned} K &= \left\{ v \in V \,|\, [v]_n \le 0 \text{ on } \Gamma_c^1 \cup \Gamma_c^2 \right\}, \\ P &= \left\{ \tau \in S \,|\, \|\tau(x)^D\|_F \le \gamma \text{ for a. a. } x \in \Omega \right\} \\ \Lambda_\lambda &= \{ \tau \in S \,|\, \langle \tau, \varepsilon(v) \rangle \ge \lambda L(v) \ \ \forall v \in K \}, \ L(v) = \int_\Omega F \cdot v \mathrm{d}x + \int_{\Gamma_f} g \cdot v \mathrm{d}s \end{aligned}$$

where $\|.\|_F$ denotes the Frobenius norm, τ^D is the deviatoric part of τ, $\gamma > 0$ represents the initial yield stress and $\lambda \geq 0$ is the above mentioned load parameter.

The dual and primal formulations of the problem in dependence on $\lambda \geq 0$ read as follows [5, 9]:

$$\begin{aligned} (\mathcal{P})^*_\lambda \qquad & \text{minimize} \qquad \mathcal{S}(\tau) = \frac{1}{2}\langle \mathbb{C}^{-1}\tau, \tau\rangle \quad \text{on } \Lambda_\lambda \cap P, \\ (\mathcal{P})_\lambda \qquad & \text{minimize} \qquad \mathcal{J}_\lambda(v) = \Psi(\varepsilon(v)) - \lambda L(v) \quad \text{on } K. \end{aligned}$$

Here the functional Ψ is convex, Fréchet differentiable and can be split into the volumetric and deviatoric part, i.e. $\Psi(e) = \Psi_V(\mathrm{tr}(e)) + \Psi_D(e^D)$ for any $e = \frac{1}{3}\mathrm{tr}(e)I + e^D \in S$. The functional Ψ_V is quadratic while Ψ_D has only a linear growth:

$$\exists k_0, k_1 > 0: \quad k_0\left(\|e\|_F - 1\right) \leq \Psi_D(e) \leq k_1\|e\|_F \quad \forall e \in S. \tag{1}$$

Due to this fact, the functional $\mathcal{J}_\lambda$ is not coercive on V in general, and consequently solvability of $(\mathcal{P})_\lambda$ is not guaranteed. On the other hand, the functional $\mathcal{S}$ is quadratic. Therefore there exists a unique solution to $(\mathcal{P})^*_\lambda$ if and only if

$$\Lambda_\lambda \cap P \neq \emptyset. \tag{2}$$

The verification of (2) however is not trivial. It is known that there exists the so-called limit load $\bar{\lambda} > 0$ (possibly $\bar{\lambda} = +\infty$) such that (2) holds if and only if $\lambda \in [0, \bar{\lambda}]$. The way how to find $\bar{\lambda}$ for perfectly plastic problems with standard boundary conditions has been proposed in [9]. Its possible numerical realization can be found in [1].

The following relationship between the dual and primal problems [9] holds:

$$\inf_{v \in K} \mathcal{J}_\lambda(v) = \sup_{\tau \in \Lambda_\lambda \cap P} \{-\mathcal{S}(\tau)\} \quad \forall \lambda \geq 0, \tag{3}$$

where we set $\sup\{-\mathcal{S}(\tau)\} = -\infty$ if $\Lambda_\lambda \cap P = \emptyset$, i.e. if $\lambda > \bar{\lambda}$. This means that $\mathcal{J}_\lambda$ is bounded from below if $\lambda \leq \bar{\lambda}$, which enables us to investigate solvability of $(\mathcal{P})_\lambda$ on the larger space $BD(\Omega)$ than V, see e.g. [9]. Notice that if $\lambda < \bar{\lambda}$ then one can prove coercivity of $\mathcal{J}_\lambda$ on the non-reflexive space $LD(\Omega) = \{v \in L^1(\Omega);\ \varepsilon_{ij}(v) \in L^1(\Omega)\}$.

If we assume that there exists a solution $u_\lambda \in V$ of $(\mathcal{P})_\lambda$, then $\sigma_\lambda = T(\varepsilon(u_\lambda))$ solves $(\mathcal{P})^*_\lambda$, where T is the Fréchet derivative of Ψ representing the stress-strain operator. It is well-known that T can be defined by a projection on the convex set P. Due to this fact, T is also Lipschitz continuous and monotone on V.

3 Notes to the discretized problem

The problem is discretized by the finite element method using piecewise linear continuous approximations of the displacement field and piecewise constant approximations of the stress and strain field. We do not investigate the influence of the domain, material and load approximation.

The primal and dual formulation of the discretized problem has the same structure as in Section 2, only the spaces V, S are now finite-dimensional. Therefore the theoretical results from Section 2 remain valid. For simplicity of notation, the primal and the dual formulation of discretized problem will be denoted again by $(\mathcal{P})_\lambda$, and $(\mathcal{P})^*_\lambda$, respectively. Since V is now finite-dimensional, we can also investigate solvability of $(\mathcal{P})_\lambda$. It holds:

i) J_λ is coercive and the solution set to $(\mathcal{P})_\lambda$ is non-empty and bounded if and only if $\lambda < \bar{\lambda}$.

ii) $(\mathcal{P})_{\bar{\lambda}}$ has a solution if and only if all solutions to $(\mathcal{P})_\lambda$, $\lambda < \bar{\lambda}$, are uniformly bounded with respect to λ. In such a case the solution set to $(\mathcal{P})_{\bar{\lambda}}$ is unbounded.

iii) For sufficiently small $\lambda > 0$, $(\mathcal{P})_\lambda$ has a unique solution which also solves the corresponding problem for elastic bodies.

It is typical in perfect plasticity to investigate a loading process up to the limit load represented by $\bar{\lambda}$, which is not a priori known. So if we increase λ, we would like to known how far we are from $\bar{\lambda}$. To do this, it could be useful to know the dependence of the work of external forces L on λ. It holds:

j) Let $0 \leq \lambda_1 < \lambda_2 \leq \bar{\lambda}$ and $(\mathcal{P})_{\lambda_2}$ has a solution. Then $L(u_{\lambda_1}) < L(u_{\lambda_2})$ for any solution u_{λ_i} to $(\mathcal{P})_{\lambda_i}$, $i = 1, 2$.

jj) Let $\alpha \geq 0$ be a given parameter. Then there exist: a unique $\lambda := \lambda(\alpha) \leq \bar{\lambda}$ and a solution u_λ to $(\mathcal{P})_\lambda$ such that $L(u_\lambda) = \alpha$.

jjj) If $\alpha \to +\infty$ then $\lambda(\alpha) \to \bar{\lambda}$.

ij) The function $\alpha \mapsto \lambda(\alpha)$ is linear for sufficiently small α (elastic branch).

Thus the parameter α representing the work of the external forces is more sensitive for controlling the loading process than λ. If the curve representing the relation between α and λ is far from the initial linear behavior, one can expect that λ is close to $\bar{\lambda}$.

4 Modified semi-smooth Newton's method for $(\mathcal{P})_\lambda$

We propose a modified semi-smooth Newton method for the primal problem. The method is modified by a damping coefficient to keep a decrease of the functional and by the regularized tangential stiffness matrices to ensure a uniform positive definiteness. Although the method is primarily formulated in displacements, the main convergence results are obtained for stress fields.

Let us recall that the stress-strain operator T is potential, Lipschitz continuous and monotone. Since V is now finite dimensional, we can define a generalized derivative $\partial T(e)$ of T at any $e \in S$ in the sense of Clark [3], and a function T^o so that $T^o(e) \in \partial T(e)$, $e \in S$. Moreover it is known that T and consequently $T(\varepsilon(.))$ are strongly semi-smooth on the finite dimensional spaces S and V, respectively [8]. It means that the following estimate holds for any $v \in V$ and any sufficiently small $w \in V$:

$$T(\varepsilon(v+w)) - T(\varepsilon(v)) - T^o(\varepsilon(v+w))\varepsilon(w) = O(|||w|||^2). \tag{4}$$

Thus it is possible to use the semi-smooth Newton method [7]. On the other hand, it is not guaranteed that $T^o(\varepsilon(.))$ is positive definite in a vicinity of a solution to $(\mathcal{P})_\lambda$ since T is only monotone. For this reason we rather propose and use the regularized operator $T^{o,\nu} := (1-\nu)T^o + \nu\mathbb{C}$, $\nu \in [0,1]$, instead of T^o. It holds that

$$\langle T^{o,\nu}(\varepsilon(v))\varepsilon(w), \varepsilon(w)\rangle \geq \nu|||w|||^2 \quad \forall v, w \in V. \tag{5}$$

For Newton-like methods in optimization problems, an approximation of a non-quadratic functional by a quadratic one is typical. In our case, the functional $\mathcal{J}_\lambda$ contains the non-quadratic

part Ψ. Its value at a solution u_λ can be approximated by some $u_\lambda^k \in K$ close to u_λ as follows:

$$\Psi(\varepsilon(u_\lambda)) \approx \Psi(\varepsilon(u_\lambda^k)) + \langle T(\varepsilon(u_\lambda^k)), \varepsilon(u_\lambda - u_\lambda^k)\rangle + \frac{1}{2}\langle T^{o,\nu}(\varepsilon(u_\lambda^k))\varepsilon(u_\lambda - u_\lambda^k), \varepsilon(u_\lambda - u_\lambda^k)\rangle. \quad (6)$$

The k-th step of the modified semismooth Newton is then defined by

$$u_\lambda^{k+1} = u_\lambda^k + \alpha_k \delta u^k \in K, \quad (7)$$

where $\delta u^k \in K_k$ minimizes the quadratic functional

$$\mathcal{J}_{\lambda,k}(\delta v) = \frac{1}{2}\langle T^{o,\nu}(\varepsilon(u_\lambda^k))\varepsilon(\delta v), \varepsilon(\delta v)\rangle - \lambda L(\delta v) + \langle T(\varepsilon(u_\lambda^k)), \varepsilon(\delta v)\rangle \quad (8)$$

on the convex set $K_k := \{\delta v \in V \mid \delta v + u_\lambda^k \in K\}$ and

$$\alpha_k = \arg \min_{\alpha \in (0,1]} \mathcal{J}_\lambda(u_\lambda^k + \alpha \delta u^k). \quad (9)$$

The algorithm is initiated by some $u_\lambda^0 \in K$. We also compute the corresponding stresses $\sigma_\lambda^k = T(\varepsilon(u_\lambda^k))$. If $\nu \in (0, 1]$, then the algorithm is well-defined and determines the descent directions δu^k. The inner problem (8) is similar to the corresponding contact problem for elastic bodies. Efficient numerical methods for such a problem will be discussed in Part II of our contribution [2].

For any $\nu \in (0, 1]$, the following convergence results can be proven:

$$\lim_{k \to +\infty} \mathcal{J}_\lambda(u_\lambda^k) = \inf_{v \in K} \mathcal{J}_\lambda(v) \quad \forall \lambda \geq 0, \quad (10)$$

$$\sigma_\lambda^k \to \sigma_\lambda \quad \forall \lambda \in [0, \bar{\lambda}], \quad \sigma_\lambda \text{ solves } (\mathcal{P})_\lambda^*, \quad (11)$$

$\{u_\lambda^k\}_k$ is bounded and its accumulation points solve $(\mathcal{P})_\lambda$ for any $\lambda \in [0, \bar{\lambda})$. From (10) and (11), we see that the algorithm generates a minimization sequence of $\mathcal{J}_\lambda$ and that convergence of stresses occurs even if the primal problem does not have a unique solution, respectively.

For a wide class of the loads and $\lambda < \bar{\lambda}$, one can intuitively assume that

$$\exists \epsilon := \epsilon(\lambda) > 0 : \quad \langle T^o(\varepsilon(u_\lambda))\varepsilon(v), \varepsilon(v)\rangle \geq \epsilon |||v|||^2 \quad \forall v \in \mathbb{V}, \quad (12)$$

where $u_\lambda \in K$ is a solution to $(\mathcal{P})_\lambda$ and $\epsilon(\lambda) \to 0_+$ as $\lambda \to +\bar{\lambda}$.

If we accept this assumption then u_λ is the unique solution to $(\mathcal{P})_\lambda$ and the following convergence results hold:

$$\|\!|u_h - u_h^{k+1}|\!\| \begin{cases} \leq \left(1 - \frac{\alpha_k \epsilon}{\epsilon + (1-\epsilon)\nu}\right) |||u_h - u_h^k|||, & \forall \nu \in (0, 1], \\ = O(|||u_h - u_h^k|||^2), & \nu = 0 \end{cases} \quad (13)$$

and

$$\lim_{k \to +\infty} \alpha_k = 1. \quad (14)$$

Thus we get local quadratic convergence for $\nu = 0$ and local linear convergence for $\nu \in (0, 1]$. However if the assumption (12) was true, we could expect that the inner problem (8) is ill-posed for $\lambda \to \bar{\lambda}$ and small ν.

5 Conclusion

In this contribution, we summarized and slightly extended the theoretical background to the contact problem for elastic-perfectly plastic bodies. We proposed the modified semismooth Newton method and studied its convergence. Parallel implementation of the problem and numerical examples are discussed in Part II of our contribution [2].

We have also investigated Uzawa's method for the corresponding augmented Lagrangian problem formulated in terms of displacement, strain and stress fields, see e.g. [4, 5, 6]. The comparison of both methods can be found in Part II [2].

Acknowledgement: This research has been done in the framework of the European Regional Development Fund in the IT4Innovation Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and the project SPOMECH - Creating a multidisciplinary R&D team for reliable solution of mechanical problems, reg. no. CZ.1.07/2.3.00/20.0070 supported by Operational Programme 'Education for competitiveness' funded by Structural Funds of the European Union and the state budget of the Czech Republic.

References

[1] A. Caboussat, R. Glowinski: *Numerical solution of a variational problem arising in stress analysis: The vector case*. Discrete Contin. Dyn. Syst. 27, 2010, pp. 1447–1472.

[2] M. Cermak, S. Sysala: *Numerical solution of contact perfectly plastic problems: part II - implementation*. In Proceedings of Seminar on Numerical Analysis, 2013.

[3] H. F. Clarke: *Optimization and nonsmooth analysis*. Wiley, New York, 1983.

[4] M. Fortin, R. Glowinski: *Augmented Lagrangian methods: applications to the numerical solution of boundary value problems*. Studies in Mathematics and its Applications, 15, North-Holland, 1983.

[5] J. Haslinger, I. Hlaváček: *Contact between elastic perfectly plastic bodies*. Appl. Math. 27 (1), 1982, pp. 27–45.

[6] J. Haslinger, I. Hlaváček, J. Nečas: *Numerical methods for unilateral problems in solid mechanics*. In Handbook of Numerical Analysis, Vol IV, Part 2, P.G. Ciarlet, J.L. Lions (eds.), North-Holland, 1996, pp. 313–485.

[7] L. Qi, J. Sun: *A nonsmooth version of Newton's method*. Mathematical Programming 58, 1993, pp. 353–367.

[8] S. Sysala: *Application of a modified semismooth Newton method to some elasto-plastic problems*. Math. Comp. Sim. 82, 2012, pp. 2004–2021.

[9] R. Temam: *Mathematical problems in plasticity*. Gauthier-Villars, Paris, 1985.

Parallel adaptive-multilevel BDDC method

J. Šístek[1,5], *B. Sousedík*[2,3], *J. Mandel*[4], *P. Burda*[5], *M. Čertíková*[5]

[1]Institute of Mathematics AS CR, Prague
[2]University of Southern California, Los Angeles
[3]Institute of Thermomechanics AS CR, Prague
[4]University of Colorado Denver
[5]Czech Technical University in Prague

1 Introduction

We combine the *Multilevel BDDC* (e.g. [2, 5]) with the adaptive selection of constraints [1, 3, 6] to obtain an implementation of the algorithm of *Adaptive-Multilevel BDDC* [7, 8]. This implementation is available as a part of the open-source parallel solver BDDCML.

2 Adaptive-multilevel BDDC

The goal of the *Adaptive BDDC* [1, 3, 4, 6] is to improve the coarse problem of BDDC so that the worst modes are eliminated from the space of admissible functions. This is achieved by solving a set of local generalized eigenproblems, one for each pair of subdomains. As has been shown in [4], this approach is able to significantly improve robustness of the BDDC method for problems with certain numerical difficulties, such as problems with strongly varying material coefficients.

The *Multilevel BDDC* (e.g. [2, 5]) aims at very large problems solved on large number of subdomains and corresponding processors. For such problems, the coarse problem becomes so large and/or fragmented, that factorization by a parallel direct method is not scalable or even possible. The main idea of *Multilevel BDDC* is to apply BDDC recursively to the arising coarse problems, introducing an approximation on each level. Consequently, the condition number worsens exponentially with each level [2].

The goal of the *Adaptive-Multilevel BDDC* [7, 8] is to enjoy the advantages of both of these approaches — the scalability of the multilevel approach and robustness of the *Adaptive BDDC*.

3 BDDCML package

The *Adaptive-Multilevel BDDC* has been recently included into our parallel solver BDDCML[4]. The BDDCML is a library for solving linear systems of algebraic equations in parallel. It is written in Fortran 95 programming language and parallelized by MPI. The library can be linked to users' applications, typically finite element packages. One step of the BDDC method is used as a preconditioner for the PCG method (for problems with symmetric positive definite matrix) or for the BICGstab method (for symmetric indefinite or general non-symmetric matrices).

[4]`http://www.math.cas.cz/~sistek/software/bddcml.html`

4 Numerical example

The performance of the method is analysed on a benchmark problem of elasticity analysis of a unit cube, which is loaded by its own weight and fixed at one vertical face. Nine stiff bars are cutting horizontally through the cube. The Young's modulus of the outer material E_1 is 10^6 times smaller than that of the bars E_2, creating contrast in coefficients $E_2/E_1 = 10^6$. In Fig. 1 (right), the (magnified) deformed shape of the cube is shown. The cube is discretized using uniform mesh of tri-linear finite elements and divided into an increasing number of subdomains. On the first level, subdomains are cubic with constant $H/h = 16$ ratio (H is the characteristic size of subdomains, h is the characteristic size of elements), see Fig. 1 left for an example of a division into 64 subdomains. On higher levels, divisions into subdomains are created automatically inside BDDCML by the METIS package, in general not preserving cubic shape of subdomains.

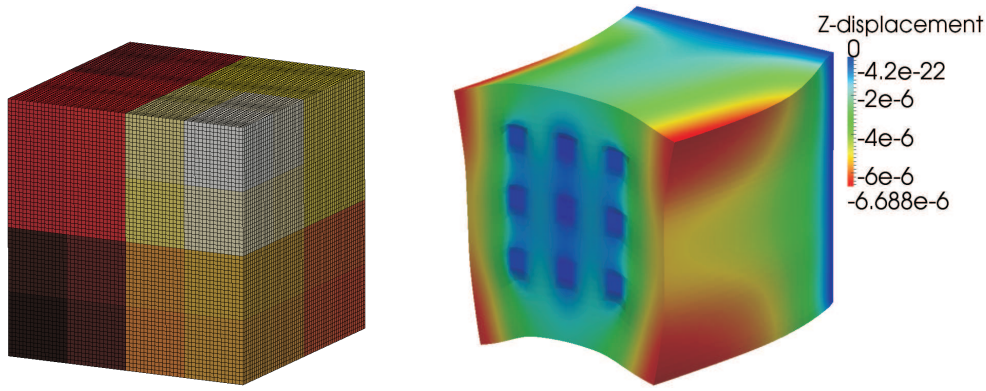


Figure 1: Example of a division of the cube into 64 subdomains (left) and (magnified) deformed shape for contrast $E_2/E_1 = 10^6$ coloured by vertical displacement (right). Reproduced from [8].

In Tabs. 1 and 2, we present results of a weak scaling test. The growing problem is solved on 8 to 32768 processors of the Cray XE6 supercomputer *Hector* (EPCC), with each core handling one subdomain of the first level. In these tables, N denotes the number of subdomains (and computer cores), n denotes global problem size, n_Γ represents the size of the reduced problem defined at the interface Γ, n_f is the number of faces in divisions on the levels (corresponding to number of generalized eigenproblems solved in the adaptive approach), 'its.' is the number of iterations needed by the PCG method, and 'cond.' is the estimated condition number obtained from the tridiagonal matrix generated in PCG. We report times needed by the set-up phase ('set-up'), by PCG iterations ('PCG') and their sum ('solve').

Results for the *non-adaptive multilevel BDDC* approach in Table 1 confirm, that convergence worsens with additional levels, as well as that the multilevel extension is capable of solving larger problems than the two-level method ('n/a' in the tables). However, we can also observe, that the non-adaptive method requires an extensive number of PCG iterations and this stage clearly dominates the overall time of solution.

The time needed by the *adaptive-multilevel BDDC* is very different. Most of it is spent by the solution of the related eigenproblems (included into time of 'set-up'). Since we keep the number of computed eigenvectors constant (ten) for each pair of subdomains, the method is not able to maintain a very low condition number after all these eigenvectors are used for generating constraints. However, number of iterations is always significantly lower than in the non-adaptive approach, and the method typically requires about one half of the computational time. While this is an important saving of computational time, it is shown (for the two-level method) in [4],

N / $\ell = 1(/2/3)$	n	n_Γ	n_f / $\ell = 1(/2/3)$	its.	cond.	time (sec) set-up	PCG	solve
2 levels								
8	0.1M	9.5k	12	582	236k	4.0	59.4	63.4
64	0.8M	0.1M	0.1k	1611	233k	4.7	171.9	176.6
512	6.4M	1.0M	1.3k	2195	240k	9.5	340.4	350.0
4096	50.9M	8.4M	11.5k	n/a	n/a	n/a	n/a	n/a
3 levels								
64/8	0.8M	0.1M	0.1k/18	2218	239k	4.7	234.1	238.8
512/64	6.4M	1.0M	1.3k/295	2830	250k	5.5	328.2	333.7
4096/512	50.9M	8.4M	11.5k/2930	4636	587k	19.3	1096.2	1115.5
32768/128	405.0M	69.1M	95.2k/664	6914	737k	155.0	3820.8	3975.8
4 levels								
512/64/8	6.4M	1.0M	1.3k/295/23	3771	729k	5.4	434.4	439.8
4096/512/64	50.9M	8.4M	11.5k/2930/380	8548	1860k	9.3	1502.3	1511.6
32768/512/8	405.0M	69.1M	95.2k/2921/23	9532	2362k	160.2	5096.6	5256.8

Table 1: Weak scaling for the cube problem with jump in coefficients $E_2/E_1 = 10^6$, *non-adaptive multilevel BDDC*. Reproduced from [8].

N / $\ell = 1(/2/3)$	n	n_Γ	n_f / $\ell = 1(/2/3)$	its.	cond.	time (sec) set-up	PCG	solve
2 levels								
8	0.1M	9.5k	12	119	1951	34.1	12.3	46.5
64	0.8M	0.1M	0.1k	76	102	96.0	8.1	104.1
512	6.4M	1.0M	1.3k	58	55	164.2	8.9	173.2
4096	50.9M	8.4M	11.5k	n/a	n/a	n/a	n/a	n/a
3 levels								
64/8	0.8M	0.1M	0.1k/18	457	48k	96.7	48.0	144.7
512/64	6.4M	1.0M	1.3k/295	82	0.1k	165.7	10.2	175.9
4096/512	50.9M	8.4M	11.5k/2930	282	165k	238.7	74.1	312.9
32768/128	405.0M	69.1M	95.2k/664	270	24k	909.4	297.6	1207.0
4 levels								
512/64/8	6.4M	1.0M	1.3k/295/23	554	63k	169.5	68.3	273.7
4096/512/64	50.9M	8.4M	11.5k/2930/380	3392	671k	299.3	800.1	1099.4
32768/512/8	405.0M	69.1M	95.2k/2921/23	3762	10495k	697.6	4925.1	5622.7

Table 2: Weak scaling for the cube problem with jump in coefficients $E_2/E_1 = 10^6$, *adaptive multilevel BDDC*. Reproduced from [8].

that the adaptive approach can solve even problems with contrasts such high, that they are not solvable by the non-adaptive approach with arithmetic averages on all faces and edges.

5 Conclusion

We have presented a recently developed parallel implementation of the Adaptive-Multilevel BDDC algorithm. The approach combines advantages of the Adaptive BDDC for numerically difficult problems and of multilevel BDDC for very large problems solved using many subdomains and cores. The developed parallel solver is available as an open-source library BDDCML.

Acknowledgement: This work was supported in part by National Science Foundation under grant DMS-1216481, by Czech Science Foundation under grant GA ČR 106/08/0403, and by the Academy of Sciences of the Czech Republic through RVO:67985840. B. Sousedík acknowledges support from the DOE/ASCR and the NSF PetaApps award number 0904754. J. Šístek acknowledges the computing time on *Hector* supercomputer provided by the PRACE-DECI initiative. A part of the work was done at the University of Colorado Denver when B. Sousedík was a graduate student and during visits of J. Šístek, partly supported by the Czech-American Cooperation program of the Ministry of Education, Youth and Sports of the Czech Republic under research project LH11004.

References

[1] J. Mandel, B. Sousedík: *Adaptive coarse space selection in the BDDC and the FETI-DP iterative substructuring methods: Optimal face degrees of freedom.* In: O. B. Widlund and D. E. Keyes, (Eds), Domain Decomposition Methods in Science and Engineering XVI, Lecture Notes in Computational Science and Engineering 55, Springer-Verlag, 2006, pp.421–428.

[2] J. Mandel, B. Sousedík, C. R. Dohrmann: *Multispace and multilevel BDDC.* Computing 83 (2–3), 2008, pp. 55–85.

[3] J. Mandel, B. Sousedík, J. Šístek: *Adaptive BDDC in three dimensions.* Math. Comput. Simulation 82 (10), 2012, pp. 1812–1831.

[4] J. Šístek, J. Mandel, B. Sousedík: *Some practical aspects of parallel adaptive BDDC method.* In: J. Brandts, J. Chleboun, S. Korotov, K. Segeth, J. Šístek, T. Vejchodský, (eds.), Proceedings of Applications of Mathematics 2012, pp. 253–266. Institute of Mathematics AS CR, 2012.

[5] J. Šístek, J. Mandel, B. Sousedík, P. Burda: *Parallel implementation of Multilevel BDDC.* In: Proceedings of ENUMATH 2011. Springer. To appear.

[6] B. Sousedík: *Comparison of some domain decomposition methods.* PhD Thesis, Czech Technical University in Prague, Faculty of Civil Engineering, Department of Mathematics, 2008.

[7] B. Sousedík: *Adaptive-Multilevel BDDC.* PhD Thesis, University of Colorado Denver, Department of Mathematical and Statistical Sciences, 2010.

[8] B. Sousedík, J. Šístek, J. Mandel: *Adaptive-Multilevel BDDC and its parallel implementation.* To appear in Computing.

On effective implementation of the non-penetration condition for non-matching grids preserving scalability of FETI based algorithms

O. Vlach, Z. Dostál, T. Kozubek, T. Brzobohatý

VŠB-Technical University of Ostrava

Mathematical models of contact include the inequalities which make the contact problems strongly nonlinear. In spite of this, a number of interesting results have been obtained by modifications of the methods that were known to be scalable for linear problems, in particular of the FETI domain decomposition method introduced by Farhat and Roux for parallel solution of linear problems. Using this approach, a body is partitioned into non-overlapping subdomains, an elliptic problem with Neumann boundary conditions is defined for each subdomain, and intersubdomain field continuity is enforced via Lagrange multipliers. The Lagrange multipliers are evaluated by solving a relatively well conditioned dual problem of small size that may be efficiently solved by a suitable variant of the conjugate gradient algorithm. Later Farhat, Mandel, and Roux [1] introduced a "natural coarse problem" whose solution was implemented by auxiliary projectors so that the resulting algorithm became scalable.

It has been soon observed that duality based domain decomposition methods may also be successful for the solution of variational inequalities that describe equilibrium of a system of elastic bodies in unilateral contact. Recently, we obtained the theoretical results that guarantee the scalability also for contact problems, see [2, 3, 4, 5] .

The scalability results were originally proved for matching grids. In this case, the boolean matrix B which imposes the "gluing" conditions and non-penetration conditions has nearly orthogonal rows, which turns out to be a key ingredient of the proofs of optimality. By nearly orthogonal we mean that the matrix B has singular values distributed in a given positive interval that does not depend on the discretization parameter. For linear problems, B can be effectively reduced to the matrix with orthogonal rows; this was used by Klawonn and Widlund to improve the estimates of the rate of convergence. The orhogonalization of constraints that they use comprises multiplication of constraints that is not admissible for inequalities that describe the non-penetrations.

The point of this paper is to extend the results mentioned above to the contact problems with non-matching grids which necessarily emerge, e.g., in the solution of transient contact problems or in contact shape optimization. We want to get both good approximation and B with nearly orthogonal rows. We consider both standard engineering approaches such as node to segment, (see Wriggers [6]) or mortar elements (see Wohlmuth or Laursen [7, 8, 9]). We give simple bounds on the singular values of the resulting matrix B and results of numerical experiments, including both the academic examples and some problems of practical interest such as the ironing eample in fig. 1. We conclude that the normalized orthogonal mortars proposed by Wohlmuth can be used to approximate the non-penetration conditions in a way that complies with the requirements of the FETI methods.

Acknowledgement: This paper has been supported by the IT4Innovations Center of Excellence project, reg. no. CZ.1.05/1.1.00/02.0070 supported by Operational Programme 'Research and Development for Innovations' funded by the Structural Funds of the European Union and the budget of the Czech Republic and by the Ministry of Education of the Czech Republic under contract No. MSM6198910027.

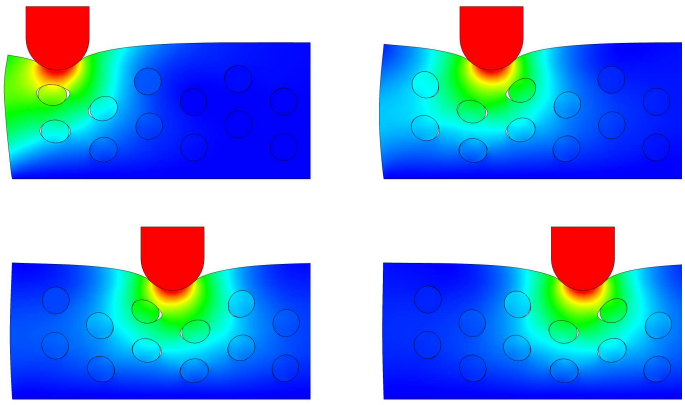


Figure 1: Ironing with insets.

References

[1] C. Farhat, J. Mandel, F.-X. Roux: *Optimal convergence properties of the FETI domain decomposition method.* Computer Methods in Applied Mechanics and Engineering 115, 1994, pp. 365–385.

[2] Z. Dostál, A. Markopoulos, T. Brzobohatý, P. Horyl, T. Kozubek: *Scalable TFETI algorithm for two dimensional multibody contact problems with friction.* Journal of Computational and Applied Mathematics 235, 2010, pp. 403–418.

[3] Z. Dostál, T. Kozubek, T. Brzobohatý, A. Markopoulos, V. Vondrák, P. Horyl: *Theoretically supported scalable TFETI algorithm for the solution of multibody 3D contact problems with friction.* Computer Methods in Applied Mechanics and Engineering, 2012, pp. 205–208, pp. 110–120.

[4] M. Sadowská, T. Kozubek, Z. Dostál, A. Markopoulos, J. Bouchala: *Scalable Total BETI based solver for 3D multibody frictionless contact problems in mechanical engineering.* Engineering Applicarions with Boundary Elements 35, 2011, pp. 330–341.

[5] Z. Dostál, T. Kozubek, T. Brzobohatý, A. Markopoulos, O. Vlach: *Scalable TFETI with preconditioning by conjugate projector for transient frictionless contact problems of elasticity.* Submitted to Computer Methods in Applied Mechanics and Engineering.

[6] P. Wriggers: *Contact mechanics.* Springer, Berlin, 2005.

[7] B. I. Wohlmuth: *Discretization methods and iterative solvers based on domain decomposition.* Lecture Notes in Computer Science and Engineering, 17, Springer, Berlin, 2001.

[8] B. I. Wohlmuth: *Variationally consistent discretization schemes and numerical algorithms for contact problems.* Acta Numerica 20, 2011, pp. 569–734.

[9] B. Yang, T. A. Laursen, X. N. Meng: *Two dimensional mortar contact methods for large deformation frictional sliding.* International Journal for Numerical Methods in Engineering 62, 2005, pp. 1183–1225.

Arbitrary accurate guaranteed bounds on homogenized coefficients by FFT-based finite element method

J. Vondřejc, J. Zeman, I. Marek

Czech Technical University in Prague

FFT-based homogenization algorithm is a popular numerical method for evaluating an effective (homogenized) matrix of periodic linear heterogeneous materials. Originally, FFT-based homogenization method was based on a solution of Lippmann-Schwinger type of integral equation with the Green function derived from an auxiliary homogeneous problem. A numerical solution proposed by Moulinec and Suquet in [1] is based on the Neumann series expansion corresponding to a simple iteration procedure.

Zeman et al. in [2] proposed a discretization of Lippmann-Schwinger equation with trigonometric collocation method by [3, 4] and showed that Moulinec-Suquet numerical algorithm corresponding to the solution of linear system can be efficiently solved by Conjugate gradients (CG) in spite of its non-symmetry, a requirement of CG to converge.

It [5, 6], it is shown that CG minimizes an energetic quadratic functional over a subspace relating to a space of curl-free fields with zero mean. Numerically, this is ensured by a projection operator deduced from Green function in Lippmann-Schwinger equation and effectively performed by Fast Fourier Transform (FFT) algorithm.

Later in [5], it has been shown that the Lippmann-Schwinger equation is equivalent to a corresponding weak formulation in a sense that the solution coincide; it also eliminates a reference homogeneous constant, a parameter of Lippmann-Schwinger equation. Next, a Galerkin approximation with numerical integration is proposed to reproduce Moulinec-Suquet algorithm; trigonometric polynomials are taken as a finite-dimensional space. Moreover, a convergence of approximate solutions to the solution of weak formulation is provided using a standard finite element approach together with estimates stated in [4].

Arbitrary precise guaranteed bounds of homogenized matrix were introduced by Dvořák in [7, 8] for a scalar problem and later independently by Wieckowski in [9] for linear elasticity. This approach is also applicable for FFT-based homogenization [10]. We present a general technique that allows effective calculations by the FFT algorithm and maintains the upper-lower bound structure. Dual formulation is applied to obtain lower bounds — for odd number of discretization points, the solution of dual formulation can be avoided. A general number of discretization points leads to a more complicated theory in both discretization and numerical treatment.

Acknowledgement: This work was supported by the Czech Science Foundation through project No. GAČR P105/12/0331.

References

[1] H. Moulinec, P. Suquet: *A fast numerical method for computing the linear and nonlinear mechanical properties of composites*. Comptes rendus de l'Académie des sciences. Série II, Mécanique, physique, chimie, astronomie 318 (11), 1994, pp. 1417–1423.

[2] J. Zeman, J. Vondřejc, J. Novák, I. Marek: *Accelerating a FFT-based solver for numerical homogenization of periodic media by conjugate gradients*. Journal of Computational Physics 229 (21), 2010, pp. 8065–8071.

[3] G. Vainikko: *Fast solvers of the Lippmann-Schwinger equation*. Direct and Inverse Problems of Mathematical Physics (R. P. Gilbert, J. Kajiwara, Y. S. Xu, eds.), International Society for Analysis, Applications and Computation, vol. 5, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000, pp. 423–440.

[4] J. Saranen, G. Vainikko: *Periodic integral and pseudodifferential equations with numerical approximation*. Springer Monographs Mathematics, 2000.

[5] J. Vondřejc, J. Zeman, I. Marek: *FFT-based finite element method homogenization*. In preparation, 2012.

[6] J. Vondřejc, J. Zeman, I. Marek: *Analysis of a fast Fourier transform based method for modeling of heterogeneous materials*. Lecture Notes in Computer Science 7116, 2012, pp. 512–522.

[7] J. Dvořák: *Optimization of composite materials*. Master's Thesis, The Charles University in Prague, June 1993.

[8] J. Dvořák: *A reliable numerical method for computing homogenized coefficients*. Technical Report.

[9] Z. Wieckowski: *Dual finite element methods in mechanics of composite materials*. Journal of Theoretical and Applied Mechanics 2 (33), pp. 233–252.

[10] J. Vondřejc, J. Zeman, I. Marek: *Guaranteed bounds of effective material properties using FFT-based FEM*. In preparation, 2012.

Winter school lectures

J. Hron
Multigrid methods for problems of mathematical physics and multiphysics

D. Lukáš
Efficient numerics for boundary integral equations

I. Marek, I. Pultarová
Algebraic multigrid, stochastic matrices and homogenization

B. Sousedík
Stochastic finite element methods

M. Vohralík
Adaptivity for linear and nonlinear solvers and time step and space mesh selection in numerical discretizations

Multigrid methods for problems of mathematical physics and multiphysics

J. Hron

Mathematical Institute, Charles University in Prague

Geometric multigrid is well established as an efficient and fast solution method for wide variety of problems. In this tutorial, we discuss development in application of geometric multigrid methods for solving linear and nonlinear systems arising from the finite element discretization of physical problems, especially solving incompressible flows. Then we discuss extension and modification of these methods to some systems describing multiphysics problems, such as bio-fluid dynamics and coupled fluid-structure interaction.

The tutorial will cover these topics:

- Introduction to classical geometric multigrid method.
- Geometric multigrid for incompressible Navier-Stokes equation.
- Extension to coupled multiphysics problems in continuum mechanics.
- Discussion of recent development and perspectives of these methods.

References

[1] W. Hackbusch: *Multi-Grid Methods and Applications*. Springer Series in Computational Mathematics 4, Springer-Verlag, Berlin, 1985.

[2] U. Trottenberg, C.W. Oosterlee, A. Schuller: *Multigrid*. Academic Press, 2001.

[3] C.C. Douglas: *Multigrid methods in science and engineering*. Computational Science & Engineering, IEEE, Vol.3, No.4, 1996, pp. 55–68.

[4] S. Turek: *Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach*. Springer, 1999.

[5] J. Hron, S. Turek: *A monolithic FEM/multigrid solver for ALE formulation of fluid-structure interaction with application in biomechanics.* In: H.-J. Bungartz (ed.): Lecture Notes in Computational Science and Engineering. Fluid–Structure Interaction – Modelling, Simulation, Optimisation. Springer-Verlag, 2006, pp. 146–170.

[6] H. Damanik, J. Hron, A. Ouazzi, S. Turek: *Monolithic Newton-multigrid solution techniques for incompressible nonlinear flow models*. Int. J. Numer. Meth. Fluids 71, 2013, pp. 208–222.

Efficient Numerics for Boundary Integral Equations

SNA '13, Rožnov pod Radhoštěm

Dalibor Lukáš

Department of Applied Mathematics, IT for Innovations
VŠB–TU Ostrava

email: dalibor.lukas@vsb.cz

Efficient Numerics for Boundary Integral Equations

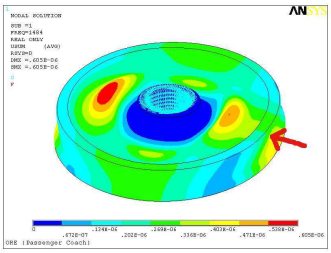
Motto

Some people like FEM, because it is easy.
Some others like BEM, because it is difficult.

Prof. Ulrich Langer, MAFELAP 2006

Efficient Numerics for Boundary Integral Equations

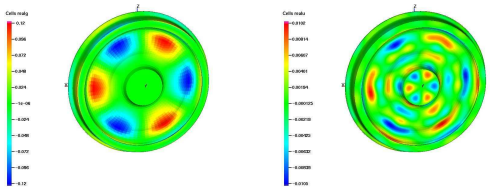
Motivation: acoustics of a railway wheel



A joint work with J. Szweda, Department of mechanics, VŠB–TU Ostrava

Efficient Numerics for Boundary Integral Equations

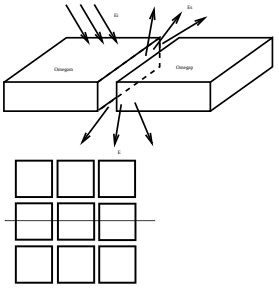
Motivation: acoustics of a railway wheel

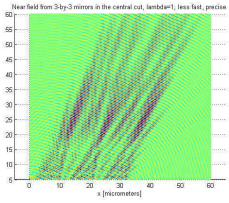


A joint work with J. Szweda, Department of mechanics, VŠB–TU Ostrava

Efficient Numerics for Boundary Integral Equations

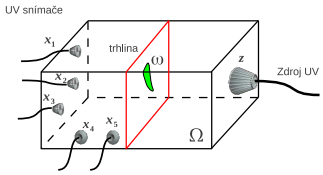
Motivation: scattering of infrared light in a DLP projector





Efficient Numerics for Boundary Integral Equations

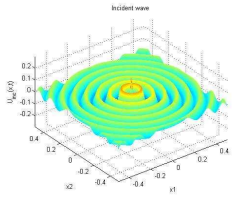
Motivation: UV deffectoscopy in aircraft industry

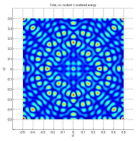


A joint TAČR project with Honeywell company.

Efficient Numerics for Boundary Integral Equations

Motivation: UV deffectoscopy in aircraft industry





A joint TAČR project with Honeywell company.

Efficient Numerics for Boundary Integral Equations

Yet, another motto

If you can't explain something simply, you don't understand it deeply enough.

A. Einstein

Efficient Numerics for Boundary Integral Equations

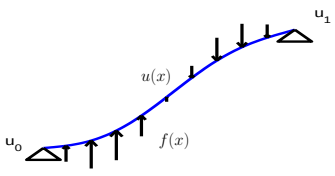
Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

1d BEM

Boundary value problem

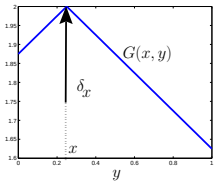
$$\begin{cases} -u''(x) = f(x), \ x \in (0,1) \\ u(0) = u_0, \\ u(1) = u_1, \end{cases}$$



1d BEM

Fundamental solution

$$G(x,y) := \begin{cases} 2 + \frac{1}{2}(y-x), & y \le x \\ 2 - \frac{1}{2}(y-x), & y \ge x \end{cases}$$



Representation theorem

For $f \in L^2(0,1)$, $x \in (0,1)$:

$$u(x) = \int_0^1 f(y)\, G(x,y)\, dy + \big[u'(y)\, G(x,y)\big]_{y=0}^{y=1} - \big[u(y)\, G'_y(x,y)\big]_{y=0}^{y=1}.$$

1d BEM

Proof of the representation theorem

For $x \in (0,1)$, $f \in L^2(0,1)$:

$$\underbrace{\int_0^1 f(y)\, G(x,y)\, dy}_{=:N(f)(x)} = \int_0^1 (-u''(y))\, G(x,y)\, dy$$

$$= \int_0^x (-u''(y))\, G(x,y)\, dy + \int_x^1 (-u''(y))\, G(x,y)\, dy$$

$$\overset{\text{per-partes}}{=} \int_0^x u'(y)\, \underbrace{G'_y(x,y)}_{=1/2}\, dy - (u'(x)\, G(x,x) - u'(0)\, G(x,0))$$

$$+ \int_x^1 u'(y)\, \underbrace{G'_y(x,y)}_{=-1/2}\, dy - (u'(1)\, G(x,1) - u'(x)\, G(x,x))$$

$$= u(x) + \underbrace{\big[u(y)\, G'_y(x,y)\big]_{y=0}^{y=1}}_{=:W(u_0,u_1)(x)} + 0 - \underbrace{\big[u'(y)\, G(x,y)\big]_{y=0}^{y=1}}_{=:\widetilde{V}(u'_0,u'_1)(x)}$$

□

1d BEM

Newton potential N

For $f \in L^2(0,1)$

$$N(f(y))(x) := \int_0^x f(y)\,\left(2+\frac{1}{2}(y-x)\right)\,dy + \int_x^1 f(y)\,\left(2-\frac{1}{2}(y-x)\right)\,dy$$

is an $H^1(0,1)$–function, which has the Dirichlet traces

$$N(f)(0) = \int_0^1 f(y)\,\left(2-\frac{y}{2}\right)\,dy, \quad N(f)(1) = \int_0^1 f(y)\,\left(2+\frac{y}{2}\right)\,dy$$

as well as the Neumann traces

$$-N(f)'(0) = N(f)'(1) = -\frac{1}{2}\int_0^1 f(y)\,dy,$$

1d BEM

Single–layer potential $\widetilde{V}$

Denote the Neumann traces of $u(x)$ by $u_0' := -u'(0)$, $u_1' := u'(1)$.

The single–layer potential is the following function

$$\widetilde{V}(u_0', u_1')(x) := \left(2-\frac{x}{2}\right)u_0' + \left(\frac{3}{2}+\frac{x}{2}\right)u_1'.$$

By applying the Dirichlet traces to the latter, we introduce the single–layer operator $V : \mathbb{R}^2 \to \mathbb{R}^2$

$$\begin{pmatrix}\widetilde{V}(u_0',u_1')(0)\\ \widetilde{V}(u_0',u_1')(0)\end{pmatrix} =: V(u_0',u_1') =: \mathbf{V}\cdot\begin{pmatrix}u_0'\\u_1'\end{pmatrix} = \begin{pmatrix}2 & \frac{3}{2}\\ \frac{3}{2} & 2\end{pmatrix}\cdot\begin{pmatrix}u_0'\\u_1'\end{pmatrix}.$$

By applying the Neumann traces to $\widetilde{V}$, we introduce the adjoint double–layer operator $K' : \mathbb{R}^2 \to \mathbb{R}^2$

$$\begin{pmatrix}-\widetilde{V}(u_0',u_1')'(0)\\ \widetilde{V}(u_0',u_1')'(1)\end{pmatrix} =: \frac{1}{2}\begin{pmatrix}u_0'\\u_1'\end{pmatrix} + K'(u_0',u_1') =: \left(\frac{1}{2}\mathbf{I}+\mathbf{K}^T\right)\cdot\begin{pmatrix}u_0'\\u_1'\end{pmatrix} := \left(\frac{1}{2}\mathbf{I}+\begin{pmatrix}0 & -\frac{1}{2}\\ -\frac{1}{2} & 0\end{pmatrix}\right)\cdot\begin{pmatrix}u_0'\\u_1'\end{pmatrix}.$$

1d BEM

Double–layer potential W

Denote the Dirichlet traces of $u(x)$ by $u_0 := u(0)$, $u_1 := u(1)$.

The double–layer potential is the following function

$$W(u_0,u_1)(x) := -\frac{1}{2}u_0 - \frac{1}{2}u_1$$

By applying the Dirichlet trace to the latter we introduce the double–layer operator $K : \mathbb{R}^2 \to \mathbb{R}^2$

$$\begin{aligned}\begin{pmatrix}W(u_0,u_1)(0)\\ W(u_0,u_1)(1)\end{pmatrix} =: -\frac{1}{2}\begin{pmatrix}u_0\\u_1\end{pmatrix} + K(u_0,u_1) &= \left(-\frac{1}{2}\mathbf{I}+\mathbf{K}\right)\cdot\begin{pmatrix}u_0\\u_1\end{pmatrix}\\ &= \left(-\frac{1}{2}\mathbf{I}+\begin{pmatrix}0 & -\frac{1}{2}\\ -\frac{1}{2} & 0\end{pmatrix}\right)\cdot\begin{pmatrix}u_0\\u_1\end{pmatrix}.\end{aligned}$$

Note that the Neumann traces of W vanish, which leads to the hypersingular operator $D(u_0,u_1) := (0,0)$.

1d BEM

Boundary (integral) equation(s)

Recall the representation formula: for $f \in L^2(0,1)$ and $x \in (0,1)$:

$$u(x) = N(f)(x) + \widetilde{V}(u_0,u_1)(x) - W(u_0',u_1')(x).$$

By applying the Dirichlet and Neumann traces to the latter, we arrive at the following boundary equations

$$\begin{aligned}\begin{pmatrix}u_0\\u_1\end{pmatrix} &= \begin{pmatrix}N(f)(0)\\N(f)(1)\end{pmatrix} + \mathbf{V}\cdot\begin{pmatrix}u_0'\\u_1'\end{pmatrix} - \left(-\frac{1}{2}\mathbf{I}+\mathbf{K}\right)\cdot\begin{pmatrix}u_0\\u_1\end{pmatrix},\\ \begin{pmatrix}u_0'\\u_1'\end{pmatrix} &= \begin{pmatrix}-N(f)'(0)\\N(f)'(1)\end{pmatrix} + \left(\frac{1}{2}\mathbf{I}+\mathbf{K}^T\right)\cdot\begin{pmatrix}u_0'\\u_1'\end{pmatrix} + \underbrace{\mathbf{D}}_{=\mathbf{0}}\cdot\begin{pmatrix}u_0\\u_1\end{pmatrix}.\end{aligned}$$

From the first row we can deduce the Steklov–Poincaré operator S mapping the Dirichlet traces to Neumann ones:

$$\begin{pmatrix}u_0'\\u_1'\end{pmatrix} = \underbrace{\mathbf{V}^{-1}\cdot\left(\frac{1}{2}\mathbf{I}+\mathbf{K}\right)}_{=:\mathbf{S}}\cdot\begin{pmatrix}u_0\\u_1\end{pmatrix} - \begin{pmatrix}N(f)(0)\\N(f)(1)\end{pmatrix} = \begin{pmatrix}1 & -1\\ -1 & 1\end{pmatrix}\cdot\begin{pmatrix}u_0\\u_1\end{pmatrix} - \begin{pmatrix}N(f)(0)\\N(f)(1)\end{pmatrix}.$$

The second row is just the Newton–Leibnitz formula: $u'(1) - u'(0) = -\int_0^1 f = \int_0^1 u''$.

Efficient Numerics for Boundary Integral Equations

Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

Fundamental solution, representation formula

Laplace equation with mixed boundary conditions

$\Omega \subset \mathbb{R}^2$ lipschitz domain, $\overline{\Gamma} := \overline{\partial\Omega} = \overline{\Gamma_{\mathrm{D}}} \cup \overline{\Gamma_{\mathrm{N}}}$, $\Gamma_{\mathrm{D}} \cap \Gamma_{\mathrm{N}} = \emptyset$

$$\begin{cases} -\triangle u(\mathbf{x}) &= 0, & \mathbf{x} \in \Omega \subset \mathbb{R}^2\\ \gamma_{\mathrm{D}}\, u(\mathbf{x}) := u(\mathbf{x}) &= g(\mathbf{x}), & \mathbf{x} \in \Gamma_{\mathrm{D}}\\ \gamma_{\mathrm{N}}\, u(\mathbf{x}) := \frac{\partial u}{\partial \mathbf{n}}(\mathbf{x}) &= h(\mathbf{x}), & \mathbf{x} \in \Gamma_{\mathrm{N}}\end{cases}$$

First and second Green's identities

$$\begin{aligned}\int_\Omega -\triangle u(\mathbf{y})\, v(\mathbf{y})\, d\mathbf{y} &= \int_\Omega \nabla u(\mathbf{y})\, \nabla v(\mathbf{y})\, d\mathbf{y} - \int_\Gamma \gamma_{\mathrm{N}} u(\mathbf{y})\, \gamma_{\mathrm{D}} v(\mathbf{y})\, dl(\mathbf{y})\\ \int_\Omega u(\mathbf{y})\, (-\triangle v(\mathbf{y}))\, d\mathbf{y} &= \int_\Omega \nabla u(\mathbf{y})\, \nabla v(\mathbf{y})\, d\mathbf{y} - \int_\Gamma \gamma_{\mathrm{D}} u(\mathbf{y})\, \gamma_{\mathrm{N}} v(\mathbf{y})\, dl(\mathbf{y})\end{aligned}$$

$$\int_\Omega u(\mathbf{y})\, (-\triangle v(\mathbf{y}))\, d\mathbf{y} = \int_\Omega \underbrace{-\triangle u(\mathbf{y})}_{=0}\, v(\mathbf{y})\, d\mathbf{y} + \int_\Gamma \gamma_{\mathrm{N}} u(\mathbf{y})\, \gamma_{\mathrm{D}} v(\mathbf{y})\, dl(\mathbf{y}) - \int_\Gamma \gamma_{\mathrm{D}} u(\mathbf{y})\, \gamma_{\mathrm{N}} v(\mathbf{y})\, dl(\mathbf{y})$$

Fundamental solution, representation formula

Second Green's identity

$$\int_\Omega u(\mathbf{y})\ (-\triangle v(\mathbf{y}))\ d\mathbf{y} = \int_\Gamma \gamma_{\mathrm{N}} u(\mathbf{y})\, \gamma_{\mathrm{D}} v(\mathbf{y})\, dl(\mathbf{y}) - \int_\Gamma \gamma_{\mathrm{D}} u(\mathbf{y})\, \gamma_{\mathrm{N}} v(\mathbf{y})\, dl(\mathbf{y})$$

Fundamental solution

$G(\mathbf{x},\mathbf{y}) := -\dfrac{1}{2\pi} \ln \|\mathbf{x}-\mathbf{y}\|$ satisfies $-\triangle_{\mathbf{y}} G(\mathbf{x},\mathbf{y}) = \delta_{\mathbf{x}}(\mathbf{y})$ in the distributional sense

Representation formula ($v(\mathbf{y}) := G(\mathbf{x},\mathbf{y})$)

$$\forall \mathbf{x} \in \Omega: \quad u(\mathbf{x}) = \int_\Gamma \gamma_{\mathrm{N}} u(\mathbf{y})\, \gamma_{\mathrm{D}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}) - \int_\Gamma \gamma_{\mathrm{D}} u(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{y}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})$$

We are left to calculate $\gamma_{\mathrm{D}} u$ on Γ_{N} and $\gamma_{\mathrm{N}} u$ on Γ_{D}.

Fundamental solution, representation formula

Representation formula

$$\forall \mathbf{x} \in \Omega: \quad u(\mathbf{x}) = \underbrace{\int_\Gamma \gamma_{\mathrm{N}} u(\mathbf{y})\, \gamma_{\mathrm{D}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})}_{=:\widetilde{V}(\gamma_{\mathrm{N}} u)} - \underbrace{\int_\Gamma \gamma_{\mathrm{D}} u(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{y}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})}_{=:W(\gamma_{\mathrm{D}} u)}$$

We are left to calculate $\gamma_{\mathrm{D}} u$ on Γ_{N} and $\gamma_{\mathrm{N}} u$ on Γ_{D}.

Potentials

- $\widetilde{V}(\gamma_{\mathrm{N}} u)$ ... single–layer potential
- $W(\gamma_{\mathrm{D}} u)$ ... double–layer potential

Efficient Numerics for Boundary Integral Equations

Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

Potentials, mapping properties

Single–layer potential

Given $w \in L^2(\Gamma)$, $\mathbf{x} \in \Omega$, the single–layer potential is the following function:

$$[\widetilde{V}(w(\mathbf{y}))](\mathbf{x}) := \int_\Gamma w(\mathbf{y})\, G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}) \in H^1(\Omega).$$

It is harmonic in Ω

$$\forall x \in \Omega: \quad -\triangle[\widetilde{V}(w)](\mathbf{x}) = 0.$$

Operator $\widetilde{V}$ can be continuously extended to $\widetilde{V} \in \mathcal{L}\left(H^{-1/2}(\Gamma), H^1(\Omega)\right)$, thus

$$V := \gamma_{\mathrm{D}} \circ \widetilde{V} \in \mathcal{L}\left(H^{-1/2}(\Gamma), H^{1/2}(\Gamma)\right).$$

Provided $\operatorname{diam}\Omega < 1$, V is $H^{-1/2}(\Gamma)$–elliptic.
For $w \in L^\infty(\Gamma)$, $\mathbf{x} \in \Gamma$ we arrive at a weakly–singular integral

$$[V(w)](\mathbf{x}) = \lim_{\varepsilon \to 0_+} \int_{\Gamma \setminus B_\varepsilon(\mathbf{x})} w(\mathbf{y})\, G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

Potentials, mapping properties

Adjoint double–layer potential

Since $\widetilde{V}$ can be continuosly extended to $\widetilde{V} \in \mathcal{L}\left(H^{-1/2}(\Gamma), H^1(\Omega)\right)$, then also

$$\gamma_{\mathrm{N}} \circ \widetilde{V} \in \mathcal{L}\left(H^{-1/2}(\Gamma), H^{-1/2}(\Gamma)\right).$$

For $w \in L^\infty(\Gamma)$ and smooth points $\mathbf{x} \in \Gamma$ it holds true that

$$\gamma_{\mathrm{N}}[\widetilde{V}(w)](\mathbf{x}) = \frac{1}{2} w(\mathbf{x}) + [K'(w)](\mathbf{x}),$$

where the latter is the adjoint double–layer potential

$$[K'(w(\mathbf{y}))](\mathbf{x}) := \lim_{\varepsilon \to 0_+} \int_{\Gamma \setminus B_\varepsilon(\mathbf{x})} w(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{x}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

It can be continuosly extended to

$$K' \in \mathcal{L}\left(H^{-1/2}(\Gamma), H^{-1/2}(\Gamma)\right).$$

Potentials, mapping properties

Double–layer potential

Given $v \in L^\infty(\Gamma)$, $\mathbf{x} \in \Omega$, the double–layer potential is the following, harmonic in Ω, function:

$$[W(v(\mathbf{y}))](\mathbf{x}) := \int_\Gamma v(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{y}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}) \in H^1(\Omega), \quad \forall x \in \Omega: \ -\triangle[W(v)](\mathbf{x}) = 0.$$

By continuous extension, $W \in \mathcal{L}\left(H^{1/2}(\Gamma), H^1(\Omega)\right)$, thus $\gamma_{\mathrm{D}} \circ W \in \mathcal{L}\left(H^{1/2}(\Gamma), H^{1/2}(\Gamma)\right)$. For $v \in L^\infty(\Gamma)$ and smooth points $\mathbf{x} \in \Gamma$ the following holds true:

$$\gamma_{\mathrm{D}}[W(v)](\mathbf{x}) = -\frac{1}{2} v(\mathbf{x}) + [K(v)](\mathbf{x}),$$

where the latter is the double–layer potential (a conflict of notation)

$$[K(v(\mathbf{y}))](\mathbf{x}) := \lim_{\varepsilon \to 0_+} \int_{\Gamma \setminus B_\varepsilon(\mathbf{x})} v(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{y}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

It can be continuously extended to $K \in \mathcal{L}\left(H^{1/2}(\Gamma), H^{1/2}(\Gamma)\right)$.

Potentials, mapping properties

Hypersingular operator

Since W can be continuously extended to $W \in \mathcal{L}\left(H^{1/2}(\Gamma), H^1(\Omega)\right)$, then also

$$\gamma_{\mathrm{N}} \circ W \in \mathcal{L}\left(H^{1/2}(\Gamma), H^{-1/2}(\Gamma)\right).$$

Operator $D := -\gamma_{\mathrm{N}} \circ W$ is called the hypersingular operator. It is not defined in terms of the Cauchy principal value. Rather, for $v \in C(\Gamma)$ it takes the form

$$[D(v)](\mathbf{x}) = -\int_\Gamma (v(\mathbf{y}) - v(\mathbf{x}))\, \gamma_{\mathrm{N},\mathbf{x}}\, \gamma_{\mathrm{N},\mathbf{y}}\, G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

For $u, v \in C(\Gamma)$ piecewise cotinuously differentiable the integration by–parts results in

$$\begin{aligned}\langle D(u), v\rangle_\Gamma &= -\int_\Gamma v(\mathbf{x})\, \gamma_{\mathrm{N},\mathbf{x}} \int_\Gamma u(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{y}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})\, dl(\mathbf{x}) \\ &= -\int_\Gamma \frac{dv}{dt}(\mathbf{x}) \int_\Gamma \frac{du}{dt}(\mathbf{y})\, G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})\, dl(\mathbf{x}).\end{aligned}$$

Provided $\operatorname{diam}\Omega < 1$, D is semi–elliptic on $H^{1/2}(\Gamma)$.

Potentials, mapping properties

Summary

Single–layer potential, $w \in L^\infty(\Gamma)$:

$$V \in \mathcal{L}\left(H^{-1/2}(\Gamma), H^{1/2}(\Gamma)\right), \quad [V(w)](\mathbf{x}) = \lim_{\varepsilon\to 0+} \int_{\Gamma\setminus B_\varepsilon(\mathbf{x})} w(\mathbf{y})\, G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

Double–layer potential, $v \in L^\infty(\Gamma)$:

$$K \in \mathcal{L}\left(H^{1/2}(\Gamma), H^{1/2}(\Gamma)\right), \quad [K(v)](\mathbf{x}) := \lim_{\varepsilon\to 0+} \int_{\Gamma\setminus B_\varepsilon(\mathbf{x})} v(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{y}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

Adjoint double–layer potential, $w \in L^\infty(\Gamma)$:

$$K' \in \mathcal{L}\left(H^{-1/2}(\Gamma), H^{-1/2}(\Gamma)\right), \quad [K'(w)](\mathbf{x}) := \lim_{\varepsilon\to 0+} \int_{\Gamma\setminus B_\varepsilon(\mathbf{x})} w(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{x}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

Hypersingular operator, $u, v \in C(\Gamma)$ piecewise continuously differentiable:

$$D \in \mathcal{L}\left(H^{1/2}(\Gamma), H^{-1/2}(\Gamma)\right), \quad \langle D(u), v\rangle_\Gamma = -\int_\Gamma \frac{dv}{dt}(\mathbf{x}) \left[V\left(\frac{du}{dt}(\mathbf{y})\right)\right](\mathbf{x})\, dl(\mathbf{x}).$$

Efficient Numerics for Boundary Integral Equations

Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

Boundary integral equations (BIE)

Representation formula

$$\forall \mathbf{x} \in \Omega: \quad u(\mathbf{x}) = \underbrace{\int_\Gamma \gamma_{\mathrm{N}} u(\mathbf{y})\, \gamma_{\mathrm{D}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})}_{=:\widetilde{V}(\gamma_{\mathrm{N}} u)} - \underbrace{\int_\Gamma \gamma_{\mathrm{D}} u(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{y}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})}_{=:W(\gamma_{\mathrm{D}} u)}$$

Boundary integral equations

Applying γ_{D} and γ_{N}, respectively, to the representation formula, the following holds true at smooth points $\mathbf{x} \in \Gamma$

$$\begin{aligned}\gamma_{\mathrm{D}} u(\mathbf{x}) &= [V(\gamma_{\mathrm{N}} u)](\mathbf{x}) + \frac{1}{2}\gamma_{\mathrm{D}} u(\mathbf{x}) - [K(\gamma_{\mathrm{D}} u)](\mathbf{x}) \\ \gamma_{\mathrm{N}} u(\mathbf{x}) &= \frac{1}{2}\gamma_{\mathrm{N}} u(\mathbf{x}) + [K'(\gamma_{\mathrm{N}} u)](\mathbf{x}) + [D(\gamma_{\mathrm{D}} u)](\mathbf{x})\end{aligned}$$

Boundary integral equations (BIE)

Representation formula

$$\forall \mathbf{x} \in \Omega: \quad u(\mathbf{x}) = \underbrace{\int_\Gamma \gamma_{\mathrm{N}} u(\mathbf{y})\, \gamma_{\mathrm{D}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})}_{=:\widetilde{V}(\gamma_{\mathrm{N}} u)} - \underbrace{\int_\Gamma \gamma_{\mathrm{D}} u(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{y}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})}_{=:W(\gamma_{\mathrm{D}} u)}$$

Boundary integral equations: weak form

Making use of the map. properties: Find $u := \gamma_{\mathrm{D}} u \in H^{1/2}(\Gamma)$, $t := \gamma_{\mathrm{N}} u \in H^{-1/2}(\Gamma)$:

$$\begin{aligned}\langle w, u\rangle &= \langle w, V(t)\rangle + \left\langle w, \left(\frac{1}{2}I - K\right)(u)\right\rangle \quad \forall w \in H^{-1/2}(\Gamma) \\ \langle t, v\rangle &= \left\langle \left(\frac{1}{2}I + K'\right)(t), v\right\rangle + \langle D(u), v\rangle \quad \forall v \in H^{1/2}(\Gamma)\end{aligned}$$

Boundary integral equations (BIE)

Direct method for the Dirichlet problem

Given a lipschitz domain $\Omega \subset \mathbb{R}^2$ and $g \in H^{1/2}(\Gamma)$.

$$\begin{cases} -\triangle u(\mathbf{x}) = 0, & \mathbf{x} \in \Omega \\ u(\mathbf{x}) = g(\mathbf{x}), & \mathbf{x} \in \Gamma \end{cases}$$

The first BIE leads to: Find $t := \gamma_{\mathrm{N}} u := \frac{du}{dn} \in H^{-1/2}(\Gamma)$:

$$\langle w, V(t)\rangle = \left\langle w, \left(\frac{1}{2}I + K\right)(g)\right\rangle \quad \forall w \in H^{-1/2}(\Gamma).$$

By Riesz theorem, it is a well–posed problem, provided $\operatorname{diam}\Omega < 1$. The volume solution reads as

$$\forall \mathbf{x} \in \Omega: \quad u(\mathbf{x}) = [\widetilde{V}(t)](\mathbf{x}) - [W(g)](\mathbf{x}).$$

Boundary integral equations (BIE)

Direct method for the Neumann problem

Given a lipschitz domain $\Omega \subset \mathbb{R}^2$ and $h \in H^{-1/2}(\Gamma)$ such that $\langle h, 1\rangle = 0$.

$$\begin{cases} -\triangle u(\mathbf{x}) &= 0, \qquad \mathbf{x} \in \Omega \\ \frac{du}{dn}(\mathbf{x}) &= h(\mathbf{x}), \ \mathbf{x} \in \Gamma \end{cases}$$

The solution is unique up to a constant. The second BIE leads to a problem with operator D, which is semi–elliptic on $H^{1/2}(\Gamma)$, provided $\operatorname{diam}\Omega < 1$. To regularize D the problem is solved in the subspace $H^{1/2}_{**}(\Gamma) := \{v \in H^{1/2}(\Gamma) : \ \langle v, 1\rangle = 0\}$.
Find $u_\alpha := \gamma_{\mathrm{D}} u \in H^{1/2}(\Gamma)$:

$$\langle D(u_\alpha), v\rangle + \alpha\langle u_\alpha, 1\rangle \langle v, 1\rangle = \left\langle \left(\frac{1}{2} I - K'\right)(h), v \right\rangle + \alpha\langle v, 1\rangle \quad \forall v \in H^{1/2}(\Gamma),$$

where $\alpha > 0$. The volume solution $u := u_\alpha + c$ is given by

$$\forall \mathbf{x} \in \Omega : \quad u_\alpha(\mathbf{x}) = [\widetilde{V}(h)](\mathbf{x}) - [W(u_\alpha)](\mathbf{x}).$$

Efficient Numerics for Boundary Integral Equations

Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

Galerkin boundary element method (BEM)

Galerkin method = orthogonal projection

Consider a Hilbert space H, a symmetric H–elliptic operator $A \in \mathcal{L}(H, H^*)$, and $b \in H^*$. We look for $u \in H$:

$$A(u) = b, \text{ i.e. } \forall v \in H : \ \langle A(u), v\rangle = \langle b, v\rangle.$$

By Riesz theorem the problem is well–posed.
Consider now a vector subspace $H^h \subset H$ and look for the Galerkin approximation $u^h \in H^h$:

$$\forall v^h \in H^h : \ \langle A(u^h), v^h\rangle = \langle b, v^h\rangle,$$

which is well–posed as well. By substracting the equations the Galerkin approximation u^h turns out to be an orthogonal projection of u

$$\forall v^h \in H^h : \quad \langle A(u - u^h), v^h\rangle = 0, \quad \text{t.j. } u^h = P_A(u).$$

Galerkin boundary element method (BEM)

Céa's lemma

$$\|u - u^h\|_H \le C \inf_{v^h \in H^h} \|u - v^h\| =: C \operatorname{dist}(u, H^h)$$

Proof. For arbitrary $v^h \in V^h$:

$$\begin{aligned} \|u - u^h\|^2_H &\overset{A \text{ ellip.}}{\le} \frac{1}{C_A}\langle A(u - u^h), u - u^h\rangle \\ &= \frac{1}{C_A}\langle A(u - u^h), u\rangle - \underbrace{\langle A(u - u^h), u^h\rangle}_{=0, \text{ see } P_A} - \underbrace{\langle A(u - u^h), v^h\rangle}_{=0, \text{ see } P_A} \\ &\overset{A \text{ bound.}}{\le} \frac{c_A}{C_A}\|u - u^h\|_H \|u - v^h\|_H \end{aligned}$$

□

Galerkin boundary element method (BEM)

2d Galerkin BEM

- Discretization: Decompose the polygonal boundary Γ into disjoint open segments

  $$\Gamma = \cup_{i=1}^n \overline{S_i}, \quad S_i \cup S_j = \emptyset \text{ for } i \ne j.$$

  Sort the segments as well as the end points in the anticlockwise order so that

  $$S_i := \{\mathbf{x}(s) := \mathbf{x}_i + (\mathbf{x}_{i+1} - \mathbf{x}_i)\, s : \ 0 < s < 1\}, \quad |S_i| := \|\mathbf{x}_{i+1} - \mathbf{x}_i\|, \quad \text{where } \mathbf{x}_{n+1} := \mathbf{x}_1.$$

- Approximate $H^{-1/2}(\Gamma)$ by L^h_0 consisting of piecewise constant functions

  $$L^h_0 := \langle \Psi_1(\mathbf{x}), \dots, \Psi_n(\mathbf{x})\rangle, \quad \text{where } \Psi_i(\mathbf{x}) := \begin{cases} 1 & \mathbf{x} \in S_i, \\ 0 & \text{elsewhere} \end{cases}$$

- Approximate $H^{1/2}(\Gamma)$ by L^h_1 consisting of continuous piecewise linear functions

  $$L^h_1 := \langle \varphi_1(\mathbf{x}), \dots, \varphi_n(\mathbf{x})\rangle, \quad \text{where } \varphi_i \in C(\Gamma), \ \varphi_i(\mathbf{x})|_{S_j} = \mathbf{a}_{ij}\cdot\mathbf{x} + b_{ij} \text{ a } \varphi_i(\mathbf{x}_j) = \delta_{ij}$$

Galerkin boundary element method (BEM)

Single–layer matrix V

Recall the 1–layer potential and the formula for $w \in L^\infty(\Gamma)$:

$$V \in \mathcal{L}\left(H^{-1/2}(\Gamma), H^{1/2}(\Gamma)\right), \quad [V(w)](\mathbf{x}) = \lim_{\varepsilon \to 0_+} \int_{\Gamma \setminus B_\varepsilon(\mathbf{x})} w(\mathbf{y})\, G(\mathbf{x}, \mathbf{y})\, dl(\mathbf{y}).$$

Thus, for $w(\mathbf{y}), z(\mathbf{x}) \in L^h_0 \subset H^{-1/2}(\Gamma)$:

$$\langle z(\mathbf{x}), V(w(\mathbf{y}))\rangle = \left\langle \sum_{i=1}^n z_i \Psi_i(\mathbf{x}), V\left(\sum_{j=1}^n w_j \Psi_j(\mathbf{y})\right)\right\rangle = \mathbf{z} \cdot \mathbf{V} \cdot \mathbf{w},$$

where

$$(\mathbf{V})_{ij} := \int_{S_i}\int_{S_j} G(\mathbf{x}, \mathbf{y})\, dl(\mathbf{y})\, dl(\mathbf{x}), \quad \mathbf{z} := (z_1, \dots, z_n), \quad \mathbf{w} := (w_1, \dots, w_n).$$

Galerkin boundary element method (BEM)

Double–layer matrix K

Recall the 2–layer potetial and the formula for $v \in L^\infty(\Gamma)$:

$$K \in \mathcal{L}\left(H^{1/2}(\Gamma), H^{1/2}(\Gamma)\right), \quad [K(v)](\mathbf{x}) := \lim_{\varepsilon \to 0_+} \int_{\Gamma \setminus B_\varepsilon(\mathbf{x})} v(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{y}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

Thus, for $v(\mathbf{y}) \in L_1^h \subset H^{1/2}(\Gamma)$ and $z(\mathbf{x}) \in L_0^h \subset H^{-1/2}(\Gamma)$:

$$\langle z(\mathbf{x}), K(v(\mathbf{y}))\rangle = \left\langle \sum_{i=1}^n z_i \Psi_i(\mathbf{x}), K\left(\sum_{j=1}^n v_j \varphi_j(\mathbf{y})\right)\right\rangle = \mathbf{z}\cdot\mathbf{K}\cdot\mathbf{v},$$

where

$$(\mathbf{K})_{ij} := \int_{S_i}\int_{S_{j-1}\cup S_j} \varphi_j(\mathbf{y}) \frac{dG}{d\mathbf{n}_\mathbf{y}}(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})\, dl(\mathbf{x}), \quad \mathbf{z} := (z_1,\ldots,z_n), \quad \mathbf{v} := (v_1,\ldots,v_n),$$

where $S_0 := S_n$.

Galerkin boundary element method (BEM)

Adjoint double–layer matrix $\mathbf{K}' = \mathbf{K}^T$

Recall the adjoint 2–layer potential and the formula for $w \in L^\infty(\Gamma)$:

$$K' \in \mathcal{L}\left(H^{-1/2}(\Gamma), H^{-1/2}(\Gamma)\right), \quad [K'(w)](\mathbf{x}) := \lim_{\varepsilon \to 0_+} \int_{\Gamma \setminus B_\varepsilon(\mathbf{x})} w(\mathbf{y})\, \gamma_{\mathrm{N},\mathbf{x}} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

Thus, for $w(\mathbf{y}) \in L_0^h \subset H^{-1/2}(\Gamma)$ and $v(\mathbf{x}) \in L_1^h \subset H^{1/2}(\Gamma)$:

$$\langle K'(w(\mathbf{y})), v(\mathbf{x})\rangle = \left\langle K'\left(\sum_{j=1}^n w_j \Psi_j(\mathbf{y})\right), \sum_{i=1}^n v_i \varphi_i(\mathbf{x})\right\rangle = \mathbf{v}\cdot\mathbf{K}^T\cdot\mathbf{w},$$

or

$$(\mathbf{K}')_{ij} := \int_{S_{i-1}\cup S_i} \varphi_i(\mathbf{x}) \int_{S_j} \frac{dG}{d\mathbf{n}_\mathbf{x}}(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})\, dl(\mathbf{x}) = \int_{S_j}\int_{S_{i-1}\cup S_i} \varphi_i(\mathbf{x}) \frac{dG}{d\mathbf{n}_\mathbf{x}}(\mathbf{x},\mathbf{y})\, dl(\mathbf{x})\, dl(\mathbf{y}) = (\mathbf{K})_{ji}.$$

Galerkin boundary element method (BEM)

Hypersingular matrix D

Recall the hypersingular operator and the formula for $u, v \in C(\Gamma)$ pcw. cont. diff.:

$$D \in \mathcal{L}\left(H^{1/2}(\Gamma), H^{-1/2}(\Gamma)\right), \quad \langle D(u), v\rangle_\Gamma = -\int_\Gamma \frac{dv}{dt}(\mathbf{x}) \left[V\left(\frac{du}{dt}(\mathbf{y})\right)\right](\mathbf{x})\, dl(\mathbf{x}).$$

Thus, for $u(\mathbf{y}), v(\mathbf{x}) \in L_1^h \subset H^{1/2}(\Gamma)$:

$$\langle D(u), v\rangle = \left\langle D\left(\sum_{j=1}^n u_i \varphi_i(\mathbf{x})\right), \sum_{i=1}^n v_i \varphi_i(\mathbf{x})\right\rangle = \mathbf{v}\cdot\mathbf{D}\cdot\mathbf{u},$$

where

$$\mathbf{D} = \mathbf{T}^T\cdot\mathbf{V}\cdot\mathbf{T}, \quad \mathbf{T}_{ij} := \frac{d\varphi_j|_{S_i}}{dt}(\mathbf{x}) = \begin{cases} -1/|S_i| & j = i+1 \text{ or } (i = n \text{ and } j = 1) \\ 1/|S_i| & j = i \\ 0 & \text{elsewhere} \end{cases}$$

Galerkin boundary element method (BEM)

Mass matrix M

Yet, for $w, t \in H^{-1/2}(\Gamma)$ and $u, v \in H^{1/2}(\Gamma)$ in BIE there are terms

$$\langle w, I(u)\rangle = \langle w, u\rangle = \int_\Gamma w(\mathbf{x})\, u(\mathbf{x})\, dl(\mathbf{x}), \quad \text{resp. } \langle I(t), v\rangle = \langle t, v\rangle = \int_\Gamma t(\mathbf{x})\, v(\mathbf{x})\, dl(\mathbf{x})$$

Thus, for $u \in L_1^h \subset H^{1/2}(\Gamma)$, $w \in L_0^h \subset H^{-1/2}(\Gamma)$:

$$\langle w, I(u)\rangle = \left\langle \sum_{i=1}^n w_i \Psi_i(\mathbf{x}) \sum_{j=1}^n u_j \varphi_j(\mathbf{x})\right\rangle = \mathbf{w}\cdot\mathbf{M}\cdot\mathbf{u},$$

where

$$(\mathbf{M})_{ij} := \int_{S_i} \varphi_j(\mathbf{x})\, dl(\mathbf{x}) = \begin{cases} |S_i|/2 & j = i \text{ or } j = i+1 \text{ or } (i = 1 \text{ and } j = n) \\ 0 & \text{elsewhere} \end{cases}$$

Galerkin boundary element method (BEM)

BIE: Galerkin formulation

Find $\mathbf{u}, \mathbf{t} \in \mathbf{R}^n$:

$$\mathbf{V}\cdot\mathbf{t} - \left(\frac{1}{2}\mathbf{M} + \mathbf{K}\right)\cdot\mathbf{u} = \mathbf{0}$$
$$\left(-\frac{1}{2}\mathbf{M} + \mathbf{K}\right)^T\cdot\mathbf{t} + \mathbf{D}\cdot\mathbf{u} = \mathbf{0}$$

Galerkin boundary element method (BEM)

Dirichlet problem

Given a polygonal domain $\Omega \subset \mathbb{R}^2$ with $\operatorname{diam}\Omega < 1$ and $g \in C(\Gamma)$.

$$\begin{cases} -\triangle u(\mathbf{x}) = 0, & \mathbf{x} \in \Omega \\ u(\mathbf{x}) = g(\mathbf{x}), & \mathbf{x} \in \Gamma \end{cases}$$

The Galerkin approximation of the first BIE leads to the linear system

$$\mathbf{V}\cdot\mathbf{t} = \left(\frac{1}{2}\mathbf{M} + \mathbf{K}\right)\cdot\mathbf{g},$$

where $g_i := g(\mathbf{x}_i)$. The Neumann datum is approximated by $t^h(\mathbf{x}) := \sum_{i=1}^n t_i \Psi_i(\mathbf{x})$ and for $\mathbf{x} \in \Omega$:

$$u^h(\mathbf{x}) = \sum_{i=1}^n t_i \int_{S_i} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}) - \sum_{i=1}^n g_i \int_{S_{i-1}\cup S_i} \varphi_i(\mathbf{y}) \frac{dG}{dn_\mathbf{y}}(\mathbf{x},\mathbf{y})\, dl(\mathbf{y}).$$

Galerkin boundary element method (BEM)

Neumann problem

Given a polygonal domain $\Omega \subset \mathbb{R}^2$ with $\operatorname{diam}\Omega < 1$ and $h \in C(\Gamma)$ such that $\langle h, 1\rangle = 0$.

$$\begin{cases} -\triangle u(\mathbf{x}) &= 0, \qquad \mathbf{x} \in \Omega \\ \frac{du}{dn}(\mathbf{x}) &= h(\mathbf{x}), \ \mathbf{x} \in \Gamma \end{cases}$$

The Galerkin approximation of the second BIE leads to the linear system

$$\left(\mathbf{D} + \alpha\,(\mathbf{M}\cdot\mathbf{1})\cdot(\mathbf{M}\cdot\mathbf{1})^T\right)\mathbf{u}_\alpha = \left(\frac{1}{2}\mathbf{M} - \mathbf{K}\right)^T \cdot \mathbf{h} + \alpha\mathbf{M}^T\cdot\mathbf{1},$$

where $h_i := h\left(\frac{\mathbf{x}_i+\mathbf{x}_{i+1}}{2}\right)$. The Dirichlet datum is approximated by $u^h(\mathbf{x}) := \sum_{i=1}^n (\mathbf{u}_\alpha)_i \varphi_i(\mathbf{x}) + c$ and for $\mathbf{x} \in \Omega$:

$$u^h(\mathbf{x}) = \sum_{i=1}^n h_i \int\limits_{S_i} G(\mathbf{x},\mathbf{y})\,dl(\mathbf{y}) - \sum_{i=1}^n (\mathbf{u}_\alpha)_i \int\limits_{S_{i-1}\cup S_i} \varphi_i(\mathbf{y})\,\frac{dG}{dn_\mathbf{y}}(\mathbf{x},\mathbf{y})\,dl(\mathbf{y}) - c\int\limits_\Gamma \frac{dG}{dn_\mathbf{y}}(\mathbf{x},\mathbf{y})\,dl(\mathbf{y}).$$

Efficient Numerics for Boundary Integral Equations

Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

Numerical quadrature of singular kernels

Three types of integrals

When evaluating the entries of $\mathbf{V}$ and $\mathbf{K}$, we deal with the following integrals:

a) Identical segments — singularity in $\{\mathbf{x}=\mathbf{y}:\ \mathbf{x},\mathbf{y}\in S_i\}$, e.g.

$$(\mathbf{V})_{i,i} = \int\limits_{S_i}\int\limits_{S_i} G(\mathbf{x},\mathbf{y})\,dl(\mathbf{y})\,dl(\mathbf{x}) = -\frac{|S_i|^2}{2\pi}\int\limits_0^1\int\limits_0^1 \ln(|S^i|\,|s-t|)\,dt\,ds.$$

b) Segments with a common node — singularity at the node, e.g.

$$(\mathbf{V})_{i-1,i} = \int\limits_{S_{i-1}}\int\limits_{S_i} G(\mathbf{x},\mathbf{y})\,dl(\mathbf{y})\,dl(\mathbf{x}) = -\frac{|S_{i-1}|\,|S_i|}{2\pi}\int\limits_0^1\int\limits_0^1 \ln\|(\mathbf{x}_{i-1}-\mathbf{x}_i)s-(\mathbf{x}_{i+1}-\mathbf{x}_i)t\|\,dt\,ds.$$

c) Disjoint segments — the kernel is a C^∞ function, e.g.

$$(\mathbf{V})_{i,j} = \int\limits_{S_i}\int\limits_{S_j} G(\mathbf{x},\mathbf{y})\,dl(\mathbf{y})\,dl(\mathbf{x}) = -\frac{|S_i|\,|S_j|}{2\pi}\int\limits_0^1\int\limits_0^1 \ln\|\mathbf{x}_i+(\mathbf{x}_{i+1}-\mathbf{x}_i)s-\mathbf{x}_j-(\mathbf{x}_{j+1}-\mathbf{x}_j)t\|\,dt\,d$$

Numerical quadrature of singular kernels

a) Numerical quadrature over identical segments

Consider the parameterization $S_i := \{\mathbf{x} := \mathbf{x}_i + (\mathbf{x}_{i+1}-\mathbf{x}_i)s:\ 0<s<1\}$

$$\begin{aligned}(\mathbf{V})_{i,i} &= \int_{S_i}\int_{S_i} G(\mathbf{x},\mathbf{y})\,dl(\mathbf{y})\,dl(\mathbf{x}) = -\frac{|S_i|^2}{2\pi}\int_0^1\int_0^1 \ln(|S^i|\,|s-t|)\,dt\,ds \\ &= -\frac{|S_i|^2}{2\pi}\left(\ln|S_i| + \int_0^1\int_0^1 \ln|s-t|\,dt\,ds\right)\end{aligned}$$

Let us substitute $z := s-t$ for t in the latter integral, divide the integration domain with respect to the singularity, and integrate by–parts:

$$\begin{aligned}&\int_0^1\int_{s-1}^s \ln|z|\,dz\,ds = \int_0^1\int_{s-1}^0 1\,\ln(-z)\,dz\,ds + \int_0^1\int_0^s 1\,\ln z\,dz\,ds \\ &= -\int_0^1\int_{s-1}^s z\,\frac{1}{z}\,dz\,ds + \int_0^1 (s\,\ln s - (s-1)\ln(1-s))\,ds + \lim_{z\to 0_-} z\,\ln(-z) - \lim_{z\to 0_+} z\,\ln z \\ &= \cdots = -\frac{3}{2}, \quad \text{thus}\end{aligned}$$

$$(\mathbf{V})_{i,i} = -\frac{|S_i|^2}{2\pi}\left(\ln|S_i| - \frac{3}{2}\right)$$

Numerical quadrature of singular kernels

a) Numerical quadrature over identical segments

The procedure can be generalized to fundamental solutions of other 2d PDE operators having the sigularity for $\mathbf{x}-\mathbf{y}=\mathbf{0}$. Parameterize S_i and denote $k(s-t) := G(\mathbf{x}_i + (\mathbf{x}_{i+1}-\mathbf{x}_i)s, \mathbf{x}_i + (\mathbf{x}_{i+1}-\mathbf{x}_i)t)$.

$$\begin{aligned}(\mathbf{V})_{i,i} &= \int_{S_i}\int_{S_i} G(\mathbf{x},\mathbf{y})\,dl(\mathbf{y})\,dl(\mathbf{x}) = -|S_i|^2\int_0^1\int_0^1 k(s-t)\,dt\,ds \\ &= -|S_i|^2\int_0^1\left(\int_{s-1}^0 k(z)\,dz + \int_0^s k(z)\,dz\right)ds \\ &= \int_0^1\int_{s-1}^0 -zk'(z)\,dz\,ds + \int_0^1\int_0^s -zk'(z)\,dz\,ds + \int_0^1 (s\,k(s)-(s-1)\,k(s-1))\,ds \\ &\quad + \lim_{z\to 0_-} z\,k(z) - \lim_{z\to 0_+} z\,k(z)\end{aligned}$$

Substituting $(s-1)\,\eta := z$, $(s-1)\,d\eta = dz$, and $s\,\eta := z$, $s\,d\eta = dz$, respectively, in the first and second integral, yields

$$\int_0^1\int_0^1 \left((s-1)^2\,\eta\,k'((s-1)\eta) - s^2\,\eta\,k'(s\eta)\right)\,d\eta\,ds.$$

Numerical quadrature of singular kernels

a) Numerical quadrature over identical segments

If the kernel $k(s-t) := G(\mathbf{x}_i + (\mathbf{x}_{i+1}-\mathbf{x}_i)s, \mathbf{x}_i + (\mathbf{x}_{i+1}-\mathbf{x}_i)t)$, having the singularity for $s-t=0$, additionaly satisfies

$$z\,k(z) \quad \text{and} \quad z\,k'(z) \quad \text{continuous at } 0,$$

then

$$(\mathbf{V})_{i,i} = \int_0^1\int_0^1 \left((s-1)^2\,\eta\,k'((s-1)\eta) - s^2\,\eta\,k'(s\eta)\right)\,d\eta\,ds + \int_0^1 (s\,k(s)-(s-1)\,k(s-1))\,ds$$

and we can employ e.g. a Gauss quadrature rule with the points $\xi_k^{(N)}$ and weights $w_k^{(N)}$, $k=1,\ldots,N$:

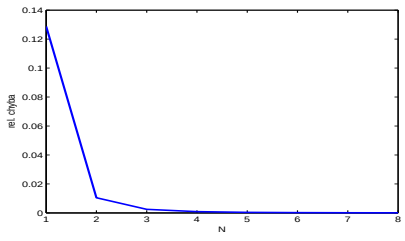
$$\begin{aligned}(\mathbf{V})_{i,i} \approx \mathbf{w}^{(N)}\cdot &\Big(\left((\xi_k^{(N)}-1)^2\,\xi_l^{(N)}\,k'((\xi_k^{(N)}-1)\xi_l^{(N)}) - (\xi_k^{(N)})^2\,\xi_l^{(N)}\,k'(\xi_k^{(N)}\xi_l^{(N)})\right)_{k,l=1}^N \cdot \mathbf{w}^{(N)} \\ &+ \left(\xi_k^{(N)}\,k(\xi_k^{(N)}) - (\xi_k^{(N)}-1)\,k(\xi_k^{(N)}-1)\right)_{k=1}^N \cdot \mathbf{w}^{(N)} =: (\mathbf{V}^{(N)})_{i,i}\end{aligned}$$

Numerical quadrature of singular kernels

a) Numerical quadrature over identical segments

Observe a numerical exponential convergence of the Gauss quadrature (to be proven).

$$\text{rel. error} := \frac{|(\mathbf{V}^{(N)})_{i,i} - (\mathbf{V})_{i,i}|}{|(\mathbf{V})_{i,i}|} \quad \text{for} \quad k(z) := -\frac{|S_i|^2}{2\pi} \ln(|S_i|\,|z|)\,.$$



Numerical quadrature of singular kernels

b) Numerical quadrature over segments with a common node

Consider the parameterization $S_{i-1} := \{\mathbf{x} := \mathbf{x}_i + (\mathbf{x}_{i-1} - \mathbf{x}_i)\, s : 0 < s < 1\}$ and $S_i := \{\mathbf{y} := \mathbf{x}_i + (\mathbf{x}_{i+1} - \mathbf{x}_i)\, t : 0 < t < 1\}$

$$(\mathbf{V})_{i-1,i} = \int_{S_{i-1}}\int_{S_i} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})\, dl(\mathbf{x}) = -\frac{|S_{i-1}|\,|S_i|}{2\pi} \int_0^1\int_0^1 \ln\|(\mathbf{x}_{i-1}-\mathbf{x}_i)s-(\mathbf{x}_{i+1}-\mathbf{x}_i)t\|\, dt\, ds.$$

The kernel $k(s,t)$ has a singularity at the origin $s = t = 0$. We replace it by decomposing the integration domain and the Duffy substitution $\tau := s$, $\tau\,\eta := p$

$$\int_0^1\int_0^1 ds\, dt = \int_0^1\int_0^t k(s,t)\, ds\, dt + \int_0^1\int_0^s k(s,t)\, dt\, ds = \int_0^1\int_0^\tau (k(\tau,p)+k(p,\tau))\, dp\, d\tau$$
$$= \int_0^1\int_0^1 \tau(k(\tau,\tau\eta)+k(\tau\eta,\tau))\, d\eta\, d\tau.$$

The resulting kernel is continuous and we can employ e.g. a Gauss quadrature

$$(\mathbf{V})_{i,j} \approx \mathbf{w}^{(N)} \cdot \left(\xi_k^{(N)}\left(k(\xi_k^{(N)}, \xi_k^{(N)}\,\xi_l^{(N)}) + k(\xi_k^{(N)}\,\xi_l^{(N)}, \xi_k^{(N)})\right)\right)_{k,l=1}^{N} \cdot \mathbf{w}^{(N)}.$$

Numerical quadrature of singular kernels

c) Numerical quadrature over disjoint segments

Consider $\overline{S_i} \cap \overline{S_j} = \emptyset$, parameterize $S_i := \{\mathbf{x} := \mathbf{x}_i + (\mathbf{x}_{i+1} - \mathbf{x}_i)\, s : 0 < s < 1\}$ and $S_j := \{\mathbf{y} := \mathbf{x}_j + (\mathbf{x}_{j+1} - \mathbf{x}_j)\, t : 0 < t < 1\}$

$$(\mathbf{V})_{i,j} = \int_{S_i}\int_{S_j} G(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})\, dl(\mathbf{x}) = -\frac{|S_i|\,|S_j|}{2\pi} \int_0^1\int_0^1 \ln\|\mathbf{x}_i+(\mathbf{x}_{i+1}-\mathbf{x}_i)s-\mathbf{x}_j-(\mathbf{x}_{j+1}-\mathbf{x}_j)t\|\, dt\, ds.$$

The kernel is a C^∞ function and the Gauss qudrature guarantees an exponential convergence

$$(\mathbf{V})_{i,j} \approx \mathbf{w}^{(N)} \cdot \left(k(\xi_k^{(N)}, \xi_l^{(N)})\right)_{k,l=1}^{N} \cdot \mathbf{w}^{(N)}.$$

Numerical quadrature of singular kernels

Numerical quadrature for K

Recall the matrix $\mathbf{K}$

$$(\mathbf{K})_{ij} := \int_{S_i}\int_{S_{j-1}\cup S_j} \varphi_j(\mathbf{y})\, \frac{dG}{d\mathbf{n}_\mathbf{y}}(\mathbf{x},\mathbf{y})\, dl(\mathbf{y})\, dl(\mathbf{x}),$$

where

$$\frac{dG}{d\mathbf{n}_\mathbf{y}}(\mathbf{x},\mathbf{y}) = \frac{1}{2\pi}\frac{(\mathbf{x}-\mathbf{y})\cdot\mathbf{n}(\mathbf{y})}{\|\mathbf{x}-\mathbf{y}\|^2}.$$

a) The identical segments, e.g. $i = j$, do not contribute, since

$$(\mathbf{x}-\mathbf{y})\cdot\mathbf{n}(\mathbf{y}) = 0.$$

b) In case of segments with a common node we employ the same technique as for $\mathbf{V}$.

c) In case of disjoint segments we employ the same technique as for $\mathbf{V}$.

Efficient Numerics for Boundary Integral Equations

Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

Matlab pseudo–code, examples

function [V, K, D, M] = BEM2dLaplace($\mathbf{X} \in \mathbb{R}^{2,n}$, $N \in \mathbb{N}$) — see a)

$\boldsymbol{\xi}^{(N,N)} := \boldsymbol{\xi}^{(N)} \bigotimes \boldsymbol{\xi}^{(N)}$, see meshgrid
$\mathbf{V} = \mathbf{K} = \mathbf{M} = \mathbf{T} := \mathbf{0}$
for $i = 1 : n$ **do**
  $\mathbf{e}_i := (i, i+1)$, or $\mathbf{e}_i := (i, 1)$ for $i = n$
  $\mathbf{x}_1 := \mathbf{X}_{:,(\mathbf{e}_i)_1}$, $\mathbf{x}_2 := \mathbf{X}_{:,(\mathbf{e}_i)_2}$
  $\boldsymbol{\tau}_i := \mathbf{x}_2 - \mathbf{x}_1$, $|S_i| := \|\boldsymbol{\tau}_i\|$
  $\mathbf{M}_{i,\mathbf{e}_i} := |S_i|\,(1/2, 1/2)$, $\mathbf{T}_{i,\mathbf{e}_i} := (-1, 1)/|S_i|$
  for $j = 1 : m$ **do**
    if $i = j$ **then**
      $\mathbf{V}_{i,i} = |S_i|^2(\ln|S_i| - 3/2)$
    else
      ...
    end if
  end for
end for
$\mathbf{V} := \mathbf{V}/(-2\pi)$, $\mathbf{K} := \mathbf{K}/(-2\pi)$, $\mathbf{D} := \mathbf{T}^T \cdot \mathbf{V} \cdot \mathbf{T}$

Matlab pseudo–code, examples

function [V, K, D, M] = BEM2dLaplace($\mathbf{X} \in \mathbb{R}^{2,n}$, $N \in \mathbb{N}$) — see b)

```
...
if i = j then
  ...
else
  e_j := (j, j+1), or e_j := (j,1) for j = n
  y_1 := X_{:,(e_j)_1}, y_2 := X_{:,(e_j)_2}
  τ_j := y_2 − y_1, |S_j| := ‖τ_j‖, n_j := ((τ_j)_2, −(τ_j)_1)/|S_j|
  if e_i ∩ e_j ≠ ∅ then
    Reorder e_i and e_j to ẽ_i and ẽ_j such that (ẽ_i)_1 = (ẽ_j)_1
    a := X_{:,(ẽ_j)_2} − X_{:,(ẽ_i)_1}, b := X_{:,(ẽ_j)_2} − X_{:,(ẽ_j)_1}
    F_1 := ((a)_1 − (b)_1 ξ^(N)).^2 + ((a)_2 − (b)_2 ξ^(N)).^2
    F_2 := ((a)_1 ξ^(N) − (b)_1).^2 + ((a)_2 ξ^(N) − (b)_2).^2, F := ln(F_1.*F_2)
    (V)_{i,j} := |S_i| |S_j| (−1 + F · w^(N))/2
    ...
  end if
end if
```

Matlab pseudo–code, examples

function [V, K, D, M] = BEM2dLaplace($\mathbf{X} \in \mathbb{R}^{2,n}$, $N \in \mathbb{N}$) — see b), c)

```
...
if e_i ∩ e_j ≠ ∅ then
  ...
  (K)_{i,(ẽ_j)_1} −= |S_i| |S_j| (a · n_j) ((1 − ξ^(N))./F_1 + (ξ^(N))./F_2) · w^(N)
  (K)_{i,(ẽ_j)_2} −= |S_i| |S_j| (a · n_j) ((ξ^(N))./F_1 + (ξ^(N))./F_2) · w^(N)
else
  a := x_1 − y_1
  A_1 := (a)_1 + (τ_i)_1 ξ^(N,N) − (τ_j)_1 ξ^(N,N)
  A_2 := (a)_2 + (τ_i)_2 ξ^(N,N) − (τ_j)_2 ξ^(N,N)
  N := A_1.^2 + A_2.^2, F := ln N, (V)_{i,j} := |S_i| |S_j| w^(N) · F · w^(N)
  F := ((n_j)_1 A_1 + (n_j)_2 A_2)./N
  K_{i,(e_j)_1} −= |S_i| |S_j| w^(N) · ((1 − ξ^(N,N)).*F) · w^(N)
  K_{i,(e_j)_2} −= |S_i| |S_j| w^(N) · (ξ^(N,N).*F) · w^(N)
end if
...
```

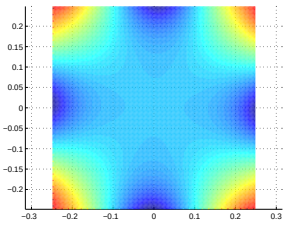
Matlab pseudo–code, examples

function [$\widetilde{\mathbf{V}}, \mathbf{W} \in \mathbb{R}^p$] = BEM2dLaplace_inner($\mathbf{X}$, N, $\mathbf{t}$, $\mathbf{u} \in \mathbb{R}^n$, $\mathbf{P} \in \mathbb{R}^{2\times p}$)

```
ξ^(N,N) := ξ^(N) ⊗ ξ^(N), see meshgrid
Ṽ = W := 0
for i = 1 : n do
  x := P_{:,i}
  for j = 1 : m do
    e_j := (j, j+1), or e_j := (j,1) for j = n, y_1 := X_{:,(e_j)_1}, y_2 := X_{:,(e_j)_2}
    τ_j := y_2 − y_1, |S_j| := ‖τ_j‖, n_j := ((τ_j)_2, −(τ_j)_1)/|S_j|
    a := x − y_1, A_1 := (a)_1 − (τ_j)_1 ξ^(N,N), A_2 := (a)_2 − (τ_j)_2 ξ^(N,N)
    N := A_1.^2 + A_2.^2, F := ln N, W̃_{:,i} += (t)_j |S_i| |S_j| w^(N) · F · w^(N)
    F := ((n_j)_1 A_1 + (n_j)_2 A_2)./N
    W̃_{:,i} −= (u)_{(e_j)_1} |S_i| |S_j| w^(N) · ((1 − ξ^(N,N)).*F) · w^(N)
    W̃_{:,i} −= (u)_{(e_j)_2} |S_i| |S_j| w^(N) · (ξ^(N,N).*F) · w^(N)
  end for
end for
Ṽ := Ṽ/(−2π), W := W/(−2π)
```

Matlab pseudo–code, examples

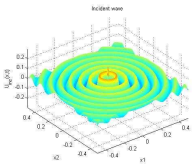
Dirichlet problem on the square $\Omega := (-1/4, 1/4)^2$, $g(\mathbf{x}) := \min_i\{|x_i|\}^2$, $n = 128$



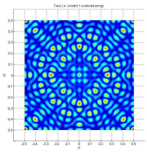
Matlab pseudo–code, examples

Neumann problem on the square

$$\begin{cases} -\triangle u(\mathbf{x}) - \kappa^2\, u(\mathbf{x}) &= -\triangle u^{\mathrm{inc}}(\mathbf{x}), & \mathbf{x} \in \Omega \subset \mathbb{R}^2 \\ \frac{du}{dn}(\mathbf{x}) &= -\frac{du^{\mathrm{inc}}}{dn}(\mathbf{x}), & \mathbf{x} \in \Gamma \end{cases} \qquad G(\mathbf{x}-\mathbf{y}) := \frac{\imath}{4} H_0^{(1)}(\kappa\|\mathbf{x}-\mathbf{y}\|)$$



incident wave u^{inc}



total field $u + u^{\mathrm{inc}}$

Matlab pseudo–code, examples

Convection–reaction–diffusion equation

$$\begin{cases} -\mathrm{div}\,(\mathbf{A}\cdot\nabla u(\mathbf{x})) + 2\mathbf{b}\cdot\nabla u(\mathbf{x}) + c\,u(\mathbf{x}) &= 0, & \mathbf{x} \in \Omega \subset \mathbb{R}^2 \\ u(\mathbf{x}) &= g(\mathbf{x}), & \mathbf{x} \in \Gamma \end{cases}$$

where $\mathbf{A} \in \mathbb{R}^{2\times 2}$ is positive definite, $\mathbf{b} \in \mathbb{R}^2$ and $c \in \mathbb{R}$ satisfy

$$c + \mathbf{b}\cdot\mathbf{A}^{-1}\cdot\mathbf{b} = 0.$$

The fundamental solution reads as follows:

$$G(\mathbf{z}) := -\frac{\mathrm{e}^{\mathbf{b}\cdot\mathbf{A}^{-1}\cdot\mathbf{z}}}{2\pi\sqrt{\det\mathbf{A}}}\,\ln\sqrt{\mathbf{z}\cdot\mathbf{A}^{-1}\cdot\mathbf{z}},$$

where $\mathbf{z} := \mathbf{x} - \mathbf{y}$.

Efficient Numerics for Boundary Integral Equations

Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

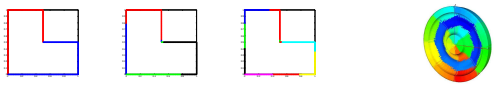
Fast BEM

Cluster geometric bisection

$C := \{\gamma_1^C, \ldots, \gamma_{n^C}^C\}$ ... cluster of elements from discretization $\{\gamma_1, \ldots, \gamma_m\}$ of Γ,
$\mathbf{x}^C := \frac{1}{\sum_k |\gamma_k^C|} \sum_k |\gamma_k^C| \, \mathbf{x}_k^C$ ... cluster centroid, where $\mathbf{x}_k^C$ is the centroid of γ_k^C,
$\mathbf{C}^C := \sum_k |\gamma_k^C| \, (\mathbf{x}_k^C - \mathbf{x}^C) \cdot (\mathbf{x}_k^C - \mathbf{x}^C)^T$ ... cluster covariance matrix,
$\mathbf{n}^C$ ... a dominant eigenvector of $\mathbf{C}^C$.

The cluster is cutted into two subclusters by the plane $(\mathbf{x} - \mathbf{x}^C) \cdot \mathbf{n}^C = 0$ as follows:
$C_1 := \left\{\gamma_k \in C \ : \ (\mathbf{x}_k^C - \mathbf{x}^C) \cdot \mathbf{n}^C \geq 0\right\}$ ... first subcluster,
$C_2 := \left\{\gamma_k \in C \ : \ (\mathbf{x}_k^C - \mathbf{x}^C) \cdot \mathbf{n}^C < 0\right\}$ ... second subcluster.

METIS could be an alternative.



Fast BEM

Admissible pairs of clusters (quadratic complexity)

$$\min\{\operatorname{diam} C_x, \operatorname{diam} C_y\} \leq \eta \operatorname{dist}(C_x, C_y), \quad \eta \in (0,1)$$

Stronger admissibility criterion (linear complexity)

$$\begin{aligned}\min\{\operatorname{diam} C_x, \operatorname{diam} C_y\} \leq \\ 2\min\{\operatorname{rad} C_x, \operatorname{rad} C_y\} \leq \eta \left(|\mathbf{x}^{C_x} - \mathbf{x}^{C_y}| - \operatorname{rad} C_x - \operatorname{rad} C_y\right) \\ \leq \eta \operatorname{dist}(C_x, C_y),\end{aligned}$$

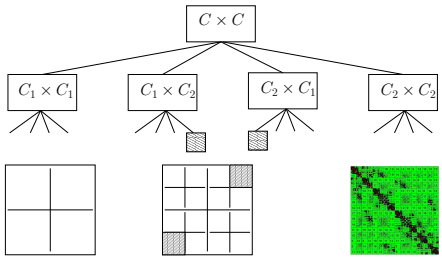
where $\operatorname{rad} C := \max_k |\mathbf{x}_k^C - \mathbf{x}^C|$.

Quad–tree of cluster pairs

$(\{\gamma_1, \ldots, \gamma_m\}, \{\gamma_1, \ldots, \gamma_m\})$ is the root.
Leaves (C, D) are either admissible or $\min\{n^C, n^D\} \leq n_{\min}$.
Nonleaves (C, D) has four sons (C_1, D_1), (C_1, D_2), (C_2, D_1), and (C_2, D_2).

Fast BEM

Quad–tree of cluster pairs, $\mathcal{H}$–matrices



Nonadmissible blocks assembled as full, admissible approximated by low–rank matrices.

Fast BEM

Compression by singular value decomposition (SVD)

$$\mathbf{A} = \sum_{i=1}^{r:=\operatorname{rank}\mathbf{A}} \sigma_i \, \mathbf{u}_i \, \mathbf{v}_i^T \approx \sum_{i=1}^{k} \sigma_i \, \mathbf{u}_i \, \mathbf{v}_i^T =: \mathbf{A}_k, \ \text{where } k < r,$$

$\sigma_1 \geq \sigma_2 \geq \ldots \sigma_r \geq 0$ ... singular values,
$(\mathbf{u}_1, \ldots, \mathbf{u}_r)$ ... an orthogonal system of left singular vectors,
$(\mathbf{v}_1, \ldots, \mathbf{v}_r)$ ... an orthogonal system of right singular vectors.
SVD gives the best approximation in the spectral (operator) norm:

$$\mathbf{A}_k = \arg \min_{\mathbf{M}:\operatorname{rank}\mathbf{M}=k} \|\mathbf{A} - \mathbf{M}\|.$$

The best compression, but worse than quadratic computational complexity $O(m\,n\,r)$.

Fast BEM

Asymptotically smooth functions

Assume $(\mathbf{A})_{i,j} := f(\mathbf{x}_i, \mathbf{y}_j)$, where $\mathbf{x}_i \in C_x$, $\mathbf{y}_j \in C_y$, $C_x, C_y \subset \mathbb{R}^d$.
$f : C_x \times C_y \to \mathbb{R}$ is asymptotically smooth if

$$\exists c_1, c_2 > 0 \ \exists g \leq 0 \ \forall \alpha \in \mathbb{N}_0^d : \ \left|\partial_{\mathbf{x}}^{\alpha} f(\mathbf{x}, \mathbf{y})\right|, \left|\partial_{\mathbf{y}}^{\alpha} f(\mathbf{x}, \mathbf{y})\right| \leq c_1 p! (c_2)^p |\mathbf{x} - \mathbf{y}|^{g-p}, \quad p = |\alpha|.$$

Compression by Taylor expansion

Provided $\operatorname{diam} C_y \leq \operatorname{diam} C_x$, $dc_2\eta < 1$, choose $\mathbf{y}_0 \in C_y$ about which we expand f:

$$\mathbf{x} \in C_x, \ \mathbf{y} \in C_y \ : \ f(\mathbf{x}, \mathbf{y}) = \sum_{k=0}^{p-1} \frac{1}{k!} ((\mathbf{y} - \mathbf{y}_0) \partial_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}_0))^k + R_p(\mathbf{x}, \mathbf{y}),$$

where

$$\begin{aligned}|R_p(\mathbf{x}, \mathbf{y})| \ &= \ \frac{1}{p!} |(\mathbf{y} - \mathbf{y}_0) \partial_{\mathbf{y}} f(\mathbf{x}, \widetilde{\mathbf{y}})|^p \leq \frac{1}{p!} d^p |\mathbf{y} - \mathbf{y}_0|^p c_1 p! (c_2)^p |\mathbf{x} - \widetilde{\mathbf{y}}|^{g-p} \\ &\leq \ c_1 d^p c_2^p \frac{\operatorname{diam}^p C_y}{\operatorname{dist}^p(C_x, C_y)} \operatorname{dist}^g(C_x, C_y) \leq c_1 (dc_2\eta)^p \operatorname{dist}^g(C_x, C_y) \to 0 \text{ as } p \to \infty.\end{aligned}$$

Function interpolation on a skeleton

Given distinct points $\mathbf{x}_1, \dots, \mathbf{x}_r \in C_x$ and $\mathbf{y}_1, \dots, \mathbf{y}_r \in C_y$.

$$f_{k+1}(\mathbf{x},\mathbf{y}) := \begin{cases} 0 & k=-1, \\ f_k(\mathbf{x},\mathbf{y}) + r_k(\mathbf{x},\mathbf{y}_{i_k})\, r_k(\mathbf{x}_{i_k},\mathbf{y}_{i_k})^{-1}\, r_k(\mathbf{x}_{i_k},\mathbf{y}) & k \geq 0 \end{cases}$$

$$r_{k+1}(\mathbf{x},\mathbf{y}) := \begin{cases} f(\mathbf{x},\mathbf{y}) & k=-1, \\ r_k(\mathbf{x},\mathbf{y}) - r_k(\mathbf{x},\mathbf{y}_{i_k})\, r_k(\mathbf{x}_{i_k},\mathbf{y}_{i_k})^{-1}\, r_k(\mathbf{x}_{i_k},\mathbf{y}) & k \geq 0 \end{cases}$$

where $\forall j \;:\; |r_k(\mathbf{x}_{i_k},\mathbf{y}_{i_k})| \geq |r_k(\mathbf{x}_{i_k},\mathbf{y}_j)| > 0$. Then $f(\mathbf{x},\mathbf{y}) = f_k(\mathbf{x},\mathbf{y}) + r_k(\mathbf{x},\mathbf{y})$ and

$$r_k(\mathbf{x}_{i_j},\mathbf{y}) = r_k(\mathbf{x},\mathbf{y}_{j_l}) = 0 \quad \text{for} \quad j,l \leq k \quad \forall \mathbf{x} \in C_x \quad \forall \mathbf{y} \in C_y.$$

Moreover, provided f asymptotically smooth, then

$$|r_{n_p}(\mathbf{x},\mathbf{y})| \leq c_1 (c_2 d\eta)^p (1+2^{n_p})\, C_p\, \mathrm{dist}^g(C_x, C_y)$$

with $n_p = \sum_{l=0}^{p-1} \binom{l+d-1}{l} \leq c_d p^d$ and C_p the Lagrange interpolation error on the skeleton.

Adaptive cross approximation (ACA)

$$\mathbf{P}_{C_x}\, \mathbf{A}\, \mathbf{P}_{C_y}^T =: \begin{pmatrix} \widetilde{\mathbf{A}}_{11} & \widetilde{\mathbf{A}}_{12} \\ \widetilde{\mathbf{A}}_{21} & \widetilde{\mathbf{A}}_{22} \end{pmatrix} \approx \begin{pmatrix} \widetilde{\mathbf{A}}_{11} & \widetilde{\mathbf{A}}_{12} \\ \widetilde{\mathbf{A}}_{21} & \widetilde{\mathbf{A}}_{21}\widetilde{\mathbf{A}}_{11}^{-1}\widetilde{\mathbf{A}}_{12} \end{pmatrix} = \begin{pmatrix} \widetilde{\mathbf{A}}_{11} \\ \widetilde{\mathbf{A}}_{21} \end{pmatrix} \left[\widetilde{\mathbf{A}}_{11}^{-1} \left(\widetilde{\mathbf{A}}_{11}, \widetilde{\mathbf{A}}_{12} \right) \right]$$
$$=: (\mathbf{u}_1, \dots, \mathbf{u}_r)\, (\mathbf{v}_1, \dots, \mathbf{v}_r)^T.$$

The rank $r := r(\varepsilon)$, where $\widetilde{\mathbf{A}}_{11} \in \mathbb{C}^{r\times r}$, is adaptively controlled by ε as follows:

$$\|\mathbf{u}_{k+1}\|_2\, \|\mathbf{v}_{k+1}\|_2 \leq \frac{\varepsilon(1-\eta)}{1+\varepsilon} \|\mathbf{A}_k\|_F, \quad \text{where } \mathbf{A}_k := \sum_{m=1}^{k} \mathbf{u}_m \mathbf{v}_m^T$$

which implies, provided $\|\mathbf{R}_{k+1}\|_F \leq \eta \|\mathbf{R}_k\|_F$, that $\frac{\|\mathbf{R}_k\|_F}{\|\mathbf{A}\|_F} \leq \varepsilon$, where $\mathbf{R}_k := \mathbf{A} - \mathbf{A}_k$. The pivots, stored in $\mathbf{P}_{C_x}$, $\mathbf{P}_{C_y}$, are chosen as to maximize $|\det \widetilde{\mathbf{A}}_{11}^k|$ with a wish to minimize $\|\mathbf{R}_k\| \equiv \|\widetilde{\mathbf{A}}_{22}^k - \widetilde{\mathbf{A}}_{21}^k (\widetilde{\mathbf{A}}_{11}^k)^{-1} \widetilde{\mathbf{A}}_{12}^k\|$.

ACA algorithm (a simple version)

Given an admissible block $\mathbf{A} \in \mathbb{C}^{m\times n}$, $\eta \in (0,1)$, and a relative precision $\varepsilon > 0$.

$k := 1$, $R := \emptyset$, $C := \emptyset$, $i_1 := 1$
repeat
  $\mathbf{v}_k := (\mathbf{A})_{i_k,*}$, $\mathbf{v}_k := \mathbf{v}_k - \sum_{l=1}^{k-1} (\mathbf{u}_l)_{i_k} \mathbf{v}_l$ % Note that $\mathbf{v}_k = (\mathbf{R}_k)_{i_k,*}$.
  $R := R \cup \{i_k\}$
  $j_k := \mathrm{argmax}_{j \notin C} |(\mathbf{v}_k)_j|$, $\mathbf{v}_k := (\mathbf{v}_k)_{j_k}^{-1} \mathbf{v}_k$
  $\mathbf{u}_k := (\mathbf{A})_{*,j_k}$, $\mathbf{u}_k := \mathbf{u}_k - \sum_{l=1}^{k-1} (\mathbf{v}_l)_{j_k} \mathbf{u}_l$ % Note that $\mathbf{u}_k = (\mathbf{R}_k)_{*,j_k}$.
  $C := C \cup \{j_k\}$
  $i_{k+1} := \mathrm{argmax}_{i \notin R} |(\mathbf{u}_k)_i|$
  $k := k+1$
until $\|\mathbf{u}_{k+1}\|_2\, \|\mathbf{v}_{k+1}\|_2 \leq \frac{\varepsilon(1-\eta)}{1+\varepsilon} \|\mathbf{A}_k\|_F$ or $R = \{1, \dots, m\}$ or $C = \{1, \dots, n\}$

The algorithm can be easily adapted to the cases $(\mathbf{v}_k)_{j_k} = 0$, $\mathbf{v}_k = \mathbf{0}$ and $\mathbf{u}_k = \mathbf{0}$.

ACA algorithm: an example ($\mathbf{R}_0 := \mathbf{A}$)

$$\mathbf{R}_0 = \begin{pmatrix} 0.431 & 0.354 & 0.582 & 0.417 \\ 0.491 & 0.396 & 0.674 & 0.449 \\ 0.446 & 0.358 & 0.583 & 0.413 \\ 0.380 & 0.328 & 0.557 & 0.372 \end{pmatrix} \xrightarrow[R=\{1\}]{i_1=1,\, j_1=3} \frac{1}{0.582} \begin{pmatrix} 0.582 \\ 0.674 \\ 0.583 \\ 0.557 \end{pmatrix} (0.431, 0.354, 0.582, 0.417)$$

$$\mathbf{R}_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -0.008 & -0.014 & 0 & -0.034 \\ 0.014 & 0.003 & 0 & -0.005 \\ -0.033 & -0.011 & 0 & -0.027 \end{pmatrix} \xrightarrow[R=\{1,2\}]{i_1=2,\, j_1=4} \frac{1}{-0.034} \begin{pmatrix} 0 \\ -0.034 \\ -0.005 \\ -0.027 \end{pmatrix} (-0.008, -0.014, 0, -0.034)$$

$$\mathbf{R}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.015 & 0.005 & 0 & 0 \\ -0.026 & 0.0004 & 0 & 0 \end{pmatrix} \xrightarrow[R=\{1,2,4\}]{i_1=4,\, j_1=1} \frac{1}{-0.026} \begin{pmatrix} 0 \\ 0 \\ 0.015 \\ -0.026 \end{pmatrix} (-0.026, 0.0004, 0, 0)$$

The relative error decays as follows: $\|\mathbf{R}_k\|_2 / \|\mathbf{A}\|_2 = 0.030, 0.016, 0.003$ for $k = 1, 2, 3$

An improved ACA for the Helmholtz equation: ball

Given the solution $p(\mathbf{x}) := \mathrm{e}^{\mathrm{i}\kappa|\mathbf{x}-\mathbf{x}_s|} / (4\pi|\mathbf{x}-\mathbf{x}_s|)$ with $\kappa := 2\pi 151.6/340$ and the scatterer placed at $\mathbf{x}_s := (0.05, 0.05, 0.05)$.

	ACA ($\eta := 0.4$, $\varepsilon := 10^{-8}$)				Elem. ACA ($\eta := 0.4$, $\varepsilon := 10^{-8}$)			
nodes/elements	error	D	GMRES	Tot. mem.	error	D	GMRES	Tot. mem.
22/40	0.099	100%/1s.	13iters./0s	7 MB	0.099	0s	13/0	8 MB
82/160	0.020	100/2	19/0	8 MB	0.020	3	19/0	9 MB
322/640	3.9e-3	100/30	28/0	13 MB	3.9e-3	34	28/1	23 MB
1282/2560	8.7e-4	94/973	42/7	85 MB	1.0e-3	255	42/49	130 MB
5122/10240	2.7e-4	39/8095	60/70	484 MB	3.9e-4	1259	60/450	727 MB
20482/40960	1.0e-4	12/42397	86/496	2.24 GB	2.3e-4	5515	86/6662	2.93 GB
81922/163840					4.2e-4	25768	123/23877	13.19 GB

An improved ACA for the Helmholtz equation: cube

Given the solution $p(\mathbf{x}) := \mathrm{e}^{\mathrm{i}\kappa|\mathbf{x}-\mathbf{x}_s|} / (4\pi|\mathbf{x}-\mathbf{x}_s|)$ with $\kappa := 2\pi 151.6/340$ and the scatterer placed at $\mathbf{x}_s := (0.05, 0.05, 0.05)$.

	ACA ($\eta := 0.4$, $\varepsilon := 10^{-8}$)				Elem. ACA ($\eta := 0.4$, $\varepsilon := 10^{-8}$)			
nodes/elements	error	D	GMRES	Tot. mem.	error	D	GMRES	Tot. mem.
8/12	0.538	100%/0s	7iters./0s	7 MB	0.538	0s	7/0	8 MB
26/48	0.186	100/0	21/0	7 MB	0.186	1	21/0	8 MB
98/192	0.166	100/3	30/0	8 MB	0.166	3	30/0	9 MB
386/768	0.035	100/43	37/1	16 MB	0.037	45	37/3	29 MB
1538/3072	9.3e-3	91/1661	48/12	115 MB	0.015	282	48/87	165 MB
6146/12288	3.5e-3	37/12134	66/107	641 MB	4.6e-3	1236	66/729	918 MB
24578/49152	3.5e-3	12/60671	92/738	2.81 GB	5.2e-3	5157	93/5060	3.90 GB

Fast BEM

An improved ACA for the Helmholtz equation: railway wheel

ACA(10^{-4},0.4)

- D: 56%/27286s,
- K: 44%, 3121s,
- GMRES: 138iters./490s

Elem. ACA(10^{-4},0.4)

- D: 78%/5732s,
- K: ?, 2658s,
- GMRES: 142iters./2524s

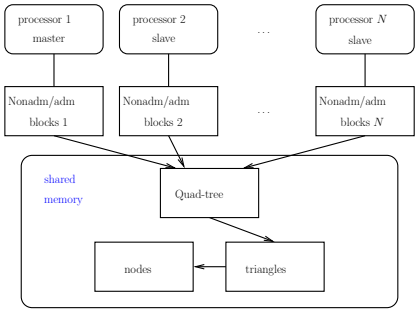
Efficient Numerics for Boundary Integral Equations

Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

Parallel BEM

Parallel implementation on a shared memory system



Parallel BEM

Helmholtz, Dirich. $u(\mathbf{x}) := e^{\imath\kappa|\mathbf{x}-\mathbf{x}_s|}/(4\pi|\mathbf{x}-\mathbf{x}_s|)$, $\kappa := 2.8$, $\mathbf{x}_s := (2,2,2)$ **on** $\mathcal{B}_1$

		compr.	scheduling+assembling times of $\mathbf{V}_\kappa$ [s]				
n	err.	of $\mathbf{V}_\kappa$	$N := 2$	$N := 4$	$N := 8$	$N := 16$	$N := 32$
40	3.3e-1	100%	0+0	0+0	0+0	0+0	0+0
160	1.2e-1	100%	0+1	0+1	0+1	0+1	0+1
640	3.6e-2	100%	0+10	0+4	0+3	0+2	0+2
2560	9.9e-3	100%	0+142	0+72	0+38	0+20	0+9
10240	2.8e-3	65%	66+1388	27+673	7+335	7+168	5+88
40960	9.0e-4	26%			452+3600	280+1823	233+929
163840	3.3e-4	8%					4011+19892

$$err. := \frac{\sqrt{\langle M(u-u_h), u-u_h\rangle_\Gamma}}{\sqrt{\langle Mu, u\rangle_\Gamma}}$$

Towards parallel scalability: $CPU = O\left(\frac{n \log n}{N}\right)$, but $Mem = O(N\, n \log n)$.

Parallel BEM

The idea

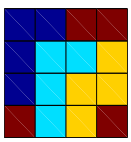
N processes, $N \times N$ submatrices

- Each diagonal block with the related geometry data assigned to one process

  $\Rightarrow$ both memory and CPU balanced, since most nonadmissible blocks are distributed efficiently.
- Each geometrically closely related $N-1$ off–diagonal blocks assigned to one process

  ? memory balanced: $Mem = O\left(\frac{n \log n}{N} + \frac{n}{\sqrt{N}}\right)$

  ? CPU balanced



Parallel BEM

Finding optimal distributions by brute force fails

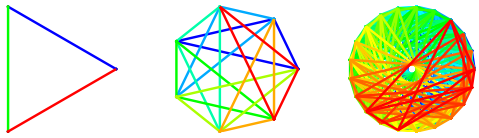
- $N = 2$: 2 cases,
- $N = 4$: 34650 cases,
- $N = 8$: $4 \cdot 10^{42}$ cases.

$$\text{number of cases} = \binom{(N-1)\,N}{N} \cdot \binom{(N-2)\,N}{N} \cdots \binom{2\,N}{N}$$

Parallel BEM

Cyclic decomposition of undirected graphs

$N := 3$ $N := 7$ $N := 21$



It is equivalent to perfect difference sets [Singer, 1934]: decompositions available for

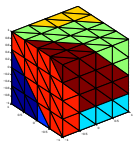
$$\frac{N(N-1)}{2N} = \frac{p(p-1)}{2},$$

where $p+1$ is a power of a prime number.

Parallel BEM

The algorithm

1. Decomposition of the mesh into N submeshes (by Metis).



2. Assignment of $O(\sqrt{N})$ submeshes to each processor, using the cyclic decomposition.
3. Parallel assembling of the $N \times N$ block matrix by means of a fast BEM.

Parallel BEM

ACA for Laplace 1-layer matrix on a cube

	compr.	average memory [MB], CPU [s] per process						
n	$\mathbf{V}$	$N := 1$	$N := 7$	$N := 31$	$N := 57$	$N := 73$	$N := 91$	$N := 133$
3072	21.5%	160, 8	148, 1	170, 0	194, 0	177, 0	197, 0	?, 0
12288	13.1%	267, 59	163, 7	175, 1	176, 1	167, 1	200, 1	207, 1
49152	5.2%	884, 367	263, 51	201, 10	194, 8	195, 6	214, 5	220, 4
196608	1.8%		705, 226	353, 53	274, 32	254, 25	280, 25	276, 18
786432	0.7%			999, 294	668, 172	599, 119	570, 110	535, 99
3145728	0.3%							1911 MB, 596 s

ACA: $\eta := 1.1$, $\varepsilon := 10^{-4}, \ldots, 10^{-9}$, $n_{\min} := 10, \ldots, 60$

Parallel scalability: $CPU = O\left(\frac{n \log n}{N}\right)$, $Mem = O\left(\frac{n \log n}{N} + \frac{n}{\sqrt{N}}\right)$.

Parallel BEM

Int. Laplace problem with Dir. datum $u(\mathbf{x}) := 1/|\mathbf{x}-(2,2,2)|$ on $\Omega := (0,1)^3$

#elems	assemble time: CPU($\mathbf{V}$)/CPU($\mathbf{K}$) [s]				
error, #CG	memory [MB] per process: compression of $\mathbf{V}$/compression of $\mathbf{K}$ [%]				
	$N := 7$	$N := 31$	$N := 57$	$N := 73$	$N := 133$
3072	114:2/84	42:2/24	24:0/21	20:1/17	
2.6e-2, 59	159:41/84	173:40/93	176:42/99	192:46/100	
12288	545:11/396	153:2/81	95:0/54	77:2/75	47:0/30
1.3e-2, 78	247:19/41	213:19/45	210:18/49	206:20/53	202:23/67
49152	2752:69/2209	819:13/474	601:6/280	446:8/292	241:7/176
6.5e-3, 102	803:8/16	347:8/17	291:8/19	277:8/20	258:9/25
196608		3171:83/2521	2122:45/1282	1885:39/1348	1016:31/790
3.3e-3, 129		1025:3/6	717:3/7	646:3/7	529:3/8
786432					4247 s:161/4085
1.7e-3, 167					1885 MB:1/3

Efficient Numerics for Boundary Integral Equations

Outline

- 1d BEM
- 2d conventional BEM
  - Fundamental solution, representation formula
  - Potentials, mapping properties
  - Boundary integral equations (BIE)
  - Galerkin boundary element method (BEM)
  - Numerical quadrature of singular kernels
  - Matlab pseudo–code, examples
- 3d fast parallel BEM
  - Fast BEM
  - Parallel BEM
- Conclusion, references

Conclusion, references

Area of use

- BEM reduces the problem to the boundary
- Fundamental solution is known for many 2d/3d PDEs, e.g., elasticity, acoustics, electromagnetism
- Recently also time–domain BEM for parabolic and hyperbolic PDEs
- Problems in bounded as well as unbounded domains
- Natural coupling with FEM
- Cons: restricted to linear material laws, difficult implementation and theory

Conclusion, references

BEM references

- Bouchala, J., Úvod do BEM. SNA 2007.
- Sadowská, M., Řešení variačních nerovnic pomocí hraničních integrálních rovnic. Diplomová práce. VŠB–TU Ostrava, 2005.
- Steinbach, O. and Rjasanow, S., The Fast Solution of Boundary Integral Equations. Springer, 2007.
- Steinbach, O., Numerical Approximation Methods for Elliptic Boundary Value Problems. Springer, 2008.
- Sauter, S. and Schwab, C., Boundary Element Methods. Springer, 2011.
- McLean, W., Strongly Elliptic Systems and Boundary Integral Equations. Cambridge University Press, 2000.
- Hsiao, G.C. and Wendland, W.L., Boundary Integral Equations. Springer, 2008.

Conclusion, references

Fast BEM references

- Bebendorf, M., Hierarchical Matrices: A Means to Efficient Solve Elliptic Boundary Value Problems. LNCSE 63, Springer, 2008.
- Nishimura, N. and Liu, Y.J., The fast multipole boundary element method for potential problems. EABE 30, 371–381, 2006.
- Greengard, L. and Rokhlin, V., A fast algorithm for particle simulations. J. Comput. Phys. 73, 1987.
- Hackbusch, W. and Nowak, Z.P., On the fast matrix multiplication in the boundary element method by panel clustering. Numer. Math. 54, 1989.

Conclusion, references

Our work

- Lukáš, D., Postava, K., and Životský, O., A shape optimization method for nonlinear axisymmetric magnetostatics using a coupling of finite and boundary elements. Math. Comp. 82, 2012.
- Lukáš, D., Kovář, P., Kovářová, T., and Merta, M., A Parallel Fast Boundary Element Method Using Cyclic Graph Decompositions. Submitted to Numer. Algorithms.

Outlook

- with M. Merta: parallel FMM BEM, SNA '13
- with L. Malý: primal BEM–based domain decomposition, SNA '13
- with A. Veit (ETH Zürich) and M. Merta: parallel BEM for the wave equation
- with P. Kovář and M. Kravčenko: (sub)optimal noncyclic decompositions of graphs

Algebraic multigrid, stochastic matrices and homogenization

I. Marek, I. Pultarová

Faculty of Civil Engineering, Czech Technical University in Prague

Multilevel and multigrid methods have become rather popular in many areas of numerical mathematics, especially in numerical solution of discretized partial differential equations (PDEs).

Theory of multigrid methods for discretized elliptic PDEs is presently quite well developed and understood. Multigrid schemes are successfully used also for the Helmholtz equation and some related theoretical results are available. Nonsymmetric problems including Markov chains have been solved by multigrid methods for several decades. Nevertheless, their theoretical justifications are still rare.

In our presentation we use the name algebraic multigrid (AMG) in cases of a solution of symmetric positive definite problems and the name iterative aggregation-disaggregation (IAD) method in case of a Markov chain. These two approaches are formally close each to the other but of course, due to the different areas of applicability, there are differences that have to be taken into account.

The presentation consists of the following main issues.

- Basic definitions and properties of Markov chains, stochastic matrices, their stationary probability distribution vectors and many related examples.
- Some areas of application of these problems.
- Difficulties that can be met during numerical computation.
- Basics of the IAD methods. Comparing them to the AMG methods.
- Theorems of convergence of the IAD methods. Emphasizing that different tools are needed than for AMG. Counter-examples of propositions that could be desired.
- Broader connections. Positive cones. Semigroups of linear operators. Partial differential operators as generators of semigroups of linear operators. A typical model problem Laplace operators.
- Homogenization.

Algebraic multigrid, stochastic matrices and homogenization

Ivo Marek, Ivana Pultarová

SNA 2013, Rožnov pod Radhoštěm

Contents

Definitions

Vector of all ones e. *Spectrum* $\sigma(M)$. *Spectral radius* $\rho(M)$.

Stochastic matrix B: $[B]_{rs} \geq 0$ and $\sum_{r=1}^{n}[B]_{rs} = 1$.

Then $||B||_1 = 1$ and thus $\rho(B) \leq 1$.

Since $e^T B = e^T$, e is left eigenvector of B, then $1 \in \sigma(B)$ and $\rho(B) = 1$.

Irreducible matrix B: there is no permutation matrix P that

$$PBP^T = \begin{pmatrix} \tilde{B}_{11} & \tilde{B}_{12} \\ 0 & \tilde{B}_{12} \end{pmatrix}.$$

In the following B will be irreducible.

Theorem. (Perron - Frobenius)
There exists a unique eigenvector $\hat{x}$ of B that $B\hat{x} = \hat{x}$ and $e^T\hat{x} = 1$.
Vector $\hat{x}$ is positive, $\hat{x} > 0$. Multiplicity of 1 in $\sigma(B)$ is one.

$\hat{x}$ is *stationary probability (distribution) vector* or *Perron eigenvector* of matrix B.

[A. Berman, R. J. Plemmons, Nonnegative Matrices in the Mathematical Sciences, 1994; R. S. Varga, Matrix Iterative Analysis, 2000]

B is *primitive* if $B^n > 0$ for some n. Otherwise B is *cyclic*.

Let $\lambda_2(B) = \max\{|\lambda|;\ \lambda \in \sigma(B), \lambda \neq 1\}$.

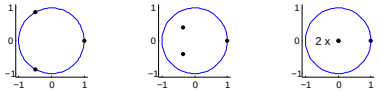
B is primitive iff $\lambda_2(B) < 1$. Then $\lim_{n\to\infty} B^n = \hat{x}e^T$. How fast?

Examples.

$$B_1 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, B_2 = \begin{pmatrix} 0 & 0.7 & 0 \\ 0.6 & 0.3 & 1 \\ 0.4 & 0 & 0 \end{pmatrix}, B_3 = \begin{pmatrix} 0.2 & 0.2 & 0.2 \\ 0.1 & 0.1 & 0.1 \\ 0.7 & 0.7 & 0.7 \end{pmatrix},$$

$$B_1^2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, B_2^2 = \begin{pmatrix} 0.42 & 0.21 & 0.7 \\ 0.58 & 0.51 & 0.3 \\ 0 & 0.28 & 0 \end{pmatrix}, B_3^2 = \begin{pmatrix} 0.2 & 0.2 & 0.2 \\ 0.1 & 0.1 & 0.1 \\ 0.7 & 0.7 & 0.7 \end{pmatrix},$$

$$B_1^3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, B_2^3 = \begin{pmatrix} 0.406 & 0.357 & 0.21 \\ 0.426 & 0.559 & 0.51 \\ 0.168 & 0.084 & 0.28 \end{pmatrix}, B_3^3 = \begin{pmatrix} 0.2 & 0.2 & 0.2 \\ 0.1 & 0.1 & 0.1 \\ 0.7 & 0.7 & 0.7 \end{pmatrix}$$



Let $P = \hat{x}e^T$ and $Z = B - P$. Then

1) $P^2 = P$, thus P is projection;

2) $e^T Z = 0$;

3) $ZP = (B-P)P = BP - P^2 = B\hat{x}e^T - P = \hat{x}e^T - P = P - P = 0$ and $PZ = 0$;

4) Eigenvalues of Z: Let $Zu = \lambda u$. Then either $\lambda = 0$ or $e^T u = 0$ (because $e^T Z = 0$). Then $Pu = 0$. Then $Bu = (P+Z)u = \lambda u$. Thus $\lambda \in \sigma(B)$.
Let $Zu = u$. Then $e^T u = 0$, thus $Pu = 0$ and thus $(P+Z)u = Bu = u$. Then u is Perron vector of B and this is the contradiction to $e^T = 0$. So that $\sigma(Z) = \sigma(B) \setminus \{1\} \cup \{0\}$.

We want to obtain an approximation to $\hat{x}$.

Let $x^0 > 0$, $e^T x^0 = 1$ and use power method: $x^{k+1} = Bx^k$. Then $e^T x^k = 1$ and

$$x^{k+1} - \hat{x} = B(x^k - \hat{x}) = (P+Z)(x^k - \hat{x}) = Z(x^k - \hat{x}).$$

The second largest eigenvalue of B is important, $\lambda_2(B)$!

Let $B\hat{x} = \hat{x}$ be

$$\begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} = \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix}.$$

Then

$$\begin{pmatrix} I - B_{11} & -B_{12} \\ -B_{21} & I - B_{22} \end{pmatrix} \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

and after elimination

$$\begin{pmatrix} I - B_{11} & -B_{12} \\ 0 & I - B_{22} - B_{21}(I - B_{11})^{-1}B_{12} \end{pmatrix} \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Then $\hat{x}_2$ is the Perron vector of the *stochastic complement of* B_{22} *in* B

$$S_{22} = B_{22} + B_{21}(I - B_{11})^{-1}B_{12}$$

which is stochastic.
Proof.

$$\begin{aligned} e^T(B_{22} + B_{21}(I - B_{11})^{-1}B_{12}) &= e^T B_{22} + e^T B_{21}(I - B_{11})^{-1}B_{12} \\ &= e^T B_{22} + e^T(I - B_{11})(I - B_{11})^{-1}B_{12} \\ &= e^T B_{22} + e^T B_{12} \\ &= e^T. \end{aligned}$$

Localization of spectrum of B

Let

$$\tau(B) := \frac{1}{2} max\{||B(e_i - e_j)||_1;\ i,j = 1,\dots,n\}$$

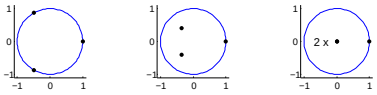
then [E. Seneta, 1984]

$$\lambda_2(B) \le \tau(B)$$

Examples.

$$B_1 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0 & 0.7 & 0 \\ 0.6 & 0.3 & 1 \\ 0.4 & 0 & 0 \end{pmatrix}, \quad B_3 = \begin{pmatrix} 0.2 & 0.2 & 0.2 \\ 0.1 & 0.1 & 0.1 \\ 0.7 & 0.7 & 0.7 \end{pmatrix},$$

$$\tau(B_1) = 1, \quad \tau(B_2) = 0.7, \quad \tau(B_3) = 0,$$
$$\lambda_2(B_1) = 1, \quad \lambda_2(B_2) \approx 0.5292, \quad \lambda_2(B_3) = 0.$$



Comparison of the upper bounds [I. Ipsen, S. Kirkland, 2006]

$$\tau(S_{22}) \le \tau(B).$$

Let A SPD, $Ax = b$ and

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{12}^T & A_{22} \end{pmatrix}$$

and the Schur complement $S_{22} = A_{22} - A_{12}^T A_{11}^{-1} A_{12}$. Then (interlacing property)

$$\lambda_{\min}(A) \le \lambda_{\min}(S_{22}) \le \lambda_{\max}(S_{22}) \le \lambda_{\max}(A).$$

Proof.

$$\frac{v^T A v}{v^T v} = \frac{v_1^T A_{11} v_1 + v_2^T A_{22} v_2 + 2 v_1^T A_{12} v_2}{v_1^T v_1 + v_2^T v_2}, \quad \frac{v_2^T S_{22} v_2}{v_2^T v_2} = \frac{v_2^T A_{22} v_2 - v_2^T A_{12}^T A_{11}^{-1} A_{12} v_2}{v_2^T v_2}.$$

First "$\le$" for $v^T = ((-A_{11}^{-1/2} A_{12} v_2)^T, v_2^T)^T$
third "$\le$" for $v^T = (0, v_2^T)^T$.

Recommendation

If B symmetric then $I - B$ positive semidefinite.

We solve here

$$(I - B)x = 0.$$

We may also solve

$$Ax = b,$$

where A is SPD.

This talk can be followed
with stochastic matrices and SPD problems in ones mind at the same time,
noticing the differences and similarities between A and $I - B$, e.g.

$$B = \begin{pmatrix} 0 & 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 & 0 \end{pmatrix}, A = I - B = \begin{pmatrix} 1 & -1/2 & 0 & -1/2 \\ -1/2 & 1 & -1/2 & 0 \\ 0 & -1/2 & 1 & -1/2 \\ -1/2 & 0 & -1/2 & 1 \end{pmatrix}$$

Applications

In general, *homogeneous discrete finite Markov chains*. Stochastic processes with discrete times $t_1, t_2, \dots$ and finite set of states $\{1, 2, \dots, N\}$.
Probability of transition from the jth state to the ith state within any time interval is constant and equal to B_{ij}.

Some applications:

1) Student's life

2) Original motivation from economy

3) Google

4) Tandem queues

5) Genetic signal processing

1) Student's life

Where the student can be found?
He can be at the university (s_1), in the library (s_2), at U Beránka (s_3) ?

Matrix of transition probabilities among states s_1, s_2, s_3

$$B = \begin{pmatrix} 0.1 & 0 & 0.4 \\ 0.4 & 0.2 & 0 \\ 0.5 & 0.8 & 0.6 \end{pmatrix}.$$

Let us have at the beginning a probability distribution v_1, then after one hour $v_2 = Bv_1$, after two hours $v_3 = Bv_2$, etc.

Thus for example, $v_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, v_2 = Bv_1 = \begin{pmatrix} 0.1 \\ 0.4 \\ 0.5 \end{pmatrix}, v_3 = Bv_2 = \begin{pmatrix} 0.21 \\ 0.12 \\ 0.67 \end{pmatrix},$

$v_4 = \begin{pmatrix} 0.289 \\ 0.108 \\ 0.603 \end{pmatrix}, v_5 = \begin{pmatrix} 0.2701 \\ 0.1372 \\ 0.5927 \end{pmatrix}, v_6 = \begin{pmatrix} 0.2666 \\ 0.1327 \\ 0.6007 \end{pmatrix}, \dots, v_\infty = \begin{pmatrix} 4/15 \\ 2/15 \\ 9/15 \end{pmatrix}$

2) Original motivation from economy

[H. A. Simon, A. Ando, Aggregation of variables in dynamic systems. Econometrica, 1961]

"Government planners are interested in the effect of a subsidy to a basic industry, say steel industry on the total effective demand in the economy."

$\Longrightarrow$ Tracing through all interactions among the economics agents,

small number of groups and separating the short-run from long-run dynamics.

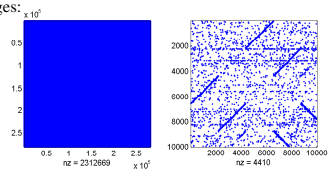
$$B = \begin{pmatrix} 0.97 & 0.02 & 0 & 0.0002 \\ 0.0291 & 0.98 & 0 & 0.0002 \\ 0.0009 & 0 & 0.96 & 0.0396 \\ 0 & 0 & 0.04 & 0.96 \end{pmatrix}, \quad \hat{x} = \begin{pmatrix} 0.1433 \\ 0.2118 \\ 0.3225 \\ 0.3225 \end{pmatrix},$$

$$y = \begin{pmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{pmatrix}, B^{100} y = \begin{pmatrix} 0.1984 \\ 0.2928 \\ 0.2549 \\ 0.2539 \end{pmatrix}, y_{new} = \begin{pmatrix} 0.1432 \\ 0.2113 \\ 0.3234 \\ 0.3222 \end{pmatrix}; B^{7000} y = \begin{pmatrix} 0.1444 \\ 0.2134 \\ 0.3211 \\ 0.3211 \end{pmatrix}$$

where $\tilde{y} = B^{100} y$ and let $\tilde{B} = RBS(\tilde{y})$ and $\tilde{B}\tilde{z} = \tilde{z}$ and $y_{new} = S(\tilde{y})\tilde{z}$.

3) Google

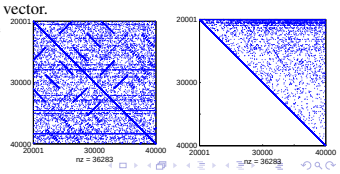
One of the ways how to evaluate the *reliability and popularity* of web pages: according to links among them.
[S. Brin, L. Page, et al., 1998;
C. Moler, The world largest computation, 2002]



Let $G_{ij} > 0$ mean that there is a link from j to i.
G huge and sparse.
Since G can be reducible, apply $B := 0.85G + 0.15ve^T$, where $v \geq 0$ and $e^T v = 1$.

Perron vector of B is the *PageRank* vector.
A higher PageRank score of a page means a higher popularity.

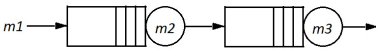
Spy-plots of Stanford web matrix
- original and its small block;
- a block and its reordering.



4) Tandem queues

System of servers connected in different ways.

For example, serial connection of two servers:
"rates of new clients coming : served at first server : served at second server ="
$= m_1 : m_2 : m_3 = 10 : 11 : 10$

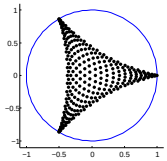


Example of $\sigma(B)$:

$N = 276$

$\lambda_2(B) = 1$

fourth largest eigenvalue 0.9890



5) Genetic signal processing

Probabilistic Boolean networks (PBN), first in [S. A. Kauffman, 1969]
Boolean network contains n elements $\{x_1, \ldots, x_n\}$, each $x_i \in \{0,1\}$,
and Boolean functions $f_i : \{0,1\}^n \to \{0,1\}$.

Genes: every $\{x_1, \ldots, x_n\}$ is a *gene activity profile* (GAP).
In case of n genes, transition matrix has $2^n \times 2^n$ elements!
Perturbations, e.g. $B_{ij} := B_{ij} + p^k$, where $p \in (0,1)$ and k is the Hamming distance of ith and jth GAPs.

Finding *optimal intervention targets*: the best gene to intervene in order to achieve the *desired attractor* (desired stable state). Mean first passage times.

"What is the probability that gene A will be expressed in the long run?"
"What is the probability that genes B and C will both be expressed in the long run?"

Most studied genes - human cancer [I. Schmulevich, 2002; M. Brun et al., 2004; W.-W. Xu et al., 2011].

Most cited (3459 refs) paper in Nature: Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls, Nature 447, (2007).

Solution methods

Basic solution methods

Let B irreducible and stochastic, $N \times N$. Solve $(I - B)x = 0$.

Direct methods

Gauss elimination

Rank of $I - B$ is $N - 1$.

Substitute some row of $I - B$ by e^T and the corresponding right hand element by 1.

Iterative methods

Krylov subspace methods

Conjugate gradient method for all extremal eigenvectors [Tanabe, 1985],

GMRES,

computed vectors can have negative elements during the computation.

Stationary matrix iterative methods

Power method

Algorithm $x^{k+1} = Bx^k$, for $x^0 > 0$, $e^T x^0 = 1$.

Error $r^{k+1} = x^{k+1} - \hat{x} = Bx^k - \hat{x} = Bx^k - B\hat{x} = Br^k$.

Denote the projection $P = \hat{x}e^T$ and $Z = B - P$. Note that $P^2 = \hat{x}e^T\hat{x}e^T = \hat{x}e^T = P$.

a) B primitive: The sequence $x^k = B^k x^0$ converges to $\hat{x}$ for any $x^0 > 0$, $e^T x_0 = 1$ and the rate of convergence is at most $\rho(Z)$.

b) B cyclic: The sequence $x^k = B^k x^0$ does not converge to $\hat{x}$ in general. The eigenvalues of B of the magnitude one are of the form $\lambda = e^{2k\pi i/m}$.

We can take $\tilde{B} = \alpha B + (1-\alpha)I$, $\alpha \in (0,1)$and use it for the iterations.
The spectrum of $\tilde{B}$ is evidently $\sigma(\tilde{B}) = 1 - \alpha + \alpha\sigma(B)$.

It is not known a priori which α is appropriate.
For sure we can always use $\tilde{B}$ instead of B for iteration.

Weak regular splitting

A is an *M-matrix* if $A = cI - B$, where $c \geq \rho(M)$.
A is a *nonsingular M-matrix* if A is M-matrix and nonsingular.
Inverse of a nonsingular M-matrix is positive, $A^{-1} > 0$.

Let M, W be a *weak regular splitting* of $I - B$,
i.e. $I - B = M - W$ and $M^{-1} \geq 0$ and $W \geq 0$.

Let $T = M^{-1}W$.

Then $T\hat{x} = \hat{x}$ and $T \geq 0$ and $\rho(T) \leq 1$.

It may happen that $\rho(B) < 1$ and $\rho(T) = 1$.

Thus $\tilde{T} = \alpha T + (1-\alpha)I$ is a good iteration matrix.

What is a suitable splitting? The choice of splitting is not straightforward:

Comparison theorems

Let A be a nonsingular M-matrix, $A = M_1 - N_1 = M_2 - N_2$ two weak regular splittings.
a) If $N_1 \geq N_2 \geq 0$ then

$$1 > \rho(M_1^{-1}N_1) \geq \rho(M_2^{-1}N_2) \geq 0.$$

b) If moreover, $A^{-1} > 0$ and $N_1 \geq N_2 \geq 0$, equality excluded, then

$$1 > \rho(M_1^{-1}N_1) > \rho(M_2^{-1}N_2) > 0.$$

Counter-example for singular M-matrices by L. Kaufman [1983]:

$$I - B = \begin{pmatrix} 1 & -1/2 & -1/2 & 0 \\ -1/2 & 1 & 0 & -1/2 \\ -1/2 & 0 & 1 & -1/2 \\ 0 & -1/2 & -1/2 & 1 \end{pmatrix},$$

$$N_1 = \begin{pmatrix} 0 & 1/2 & 1/2 & 0 \\ 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad \text{and} \quad N_2 = \begin{pmatrix} 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Then $N_1 > N_2$ but $\lambda_2(M_1^{-1}N_1) = 0$ and $\lambda_2(M_2^{-1}N_2) = 1/9$.
Different cones are needed [I. Marek, D. Szyld, 2000].

Block methods with overlap - Schwarz methods, additive and multiplicative

Restriction matrices corresponding to i-th group R_i.
Algorithm is

$$x^{k+1} = Tx^k + c,$$

where for *multiplicative Schwarz method*

$$T = \Pi_{i=1}^{p} \left(I - R_i^T (R_i A R_i^T)^{-1} R_i A \right)$$

and for *additove Schwarz method*

$$T = I - \Theta \sum_{i=1}^{p} R_i^T (R_i A R_i^T)^{-1} R_i A, \qquad \Theta > 1/p,$$

and vectors c are appropriate residual vectors.

Choice of blocks - mostly according to strength of connections or to the nonzero pattern of diagonal blocks [T. Dayar, G. Noyan, 2011]

Restricted additive Schwarz method [M. Benzi, V. Kuhlemann, 2011]. Special ordering suggested by A. Langville and C. D. Meyer, [2005, 2006]. Special methods - based on stochastic complement [C. D. Meyer, 1989].

Iterative aggregation - disaggregation (IAD) methods

Stationary matrix iterations (or Krylov subspace method) + coarse correction - recursively repeated

Building the *coarse problem*:
Reduction matrix R and *prolongation matrix* $S(y)$ are

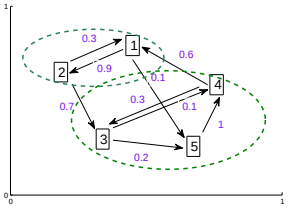
$$R = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix}, \qquad S(y) = \begin{pmatrix} 1/3 & 0 \\ 2/3 & 0 \\ 0 & 2/6 \\ 0 & 3/6 \\ 0 & 1/6 \end{pmatrix} \quad \text{for} \quad y = \begin{pmatrix} 2/12 \\ 4/12 \\ 2/12 \\ 3/12 \\ 1/12 \end{pmatrix},$$

Note, $S(y) \neq R^T$.

Identity $RS(y) = I$,

Projection $P(y) := S(y)R =$

$$= \begin{pmatrix} 1/3 & 1/3 & 0 & 0 & 0 \\ 2/3 & 2/3 & 0 & 0 & 0 \\ 0 & 0 & 2/6 & 2/6 & 2/6 \\ 0 & 0 & 3/6 & 3/6 & 3/6 \\ 0 & 0 & 1/6 & 1/6 & 1/6 \end{pmatrix}$$



Matrix $RBS(y)$ is stochastic and irreducible.

$RBS(y) =$

$$= \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0.3 & 0 & 0.6 & 0 \\ 0.9 & 0 & 0 & 0 & 0 \\ 0 & 0.7 & 0.7 & 0.3 & 0 \\ 0 & 0 & 0.1 & 0.1 & 1 \\ 0.1 & 0 & 0.2 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1/3 & 0 \\ 2/3 & 0 \\ 0 & 2/6 \\ 0 & 3/6 \\ 0 & 1/6 \end{pmatrix} =$$

$$= \begin{pmatrix} 1/2 & 3/10 \\ 1/2 & 7/10 \end{pmatrix}.$$

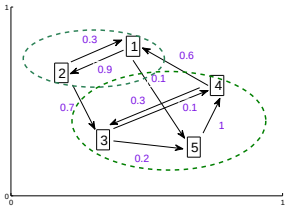
Main idea of IAD:

If $y = \hat{x}$,

the eigenvector z of $RBS(\hat{x})z = z$

is $z = R\hat{x}$

and $S(\hat{x})z = \hat{x}$!



IAD algorithm

IAD procedure (input B, y; output $\tilde{y}$)

1. μ steps of basic iteration $y := T^{\mu} y$
2. if $size(B) < \tau$ solve $RBS(y)z = z$, $e^T z = 1$,

   else **IAD procedure** (input $RBS(y)$, Ry; output z)
3. set $y := S(y)z$,

   ν steps of basic iteration $\tilde{y} := T^{\nu} y$

Some special two-level IAD methods

Koury-McAllister-Stewart method:

$\mu_1 + \nu_1 = 1$, T corresponds to block Gauss-Seidel method

Takahashi method:

modified block Gauss-Seidel m. with a coarse correction
(after recomputing of i-th part of x^k, it is normalized, $||x_i^k||_1 = [z]_k$, where z is the current solution of the coarse problem).

Vantilborgh method:

modified Jacobi method with coarse correction
(individual parts of x^k are obtained as Perron vectors of stochastic complement matrices S_{jj} of B_{jj} in $\tilde{B}$, where $\tilde{B}$ arises form B after aggregation of all blocks except the j-th one nto a single state.

[W. J. Stewart, 1994]

Convergence of the IAD methods

Let us distinguish *local* and *global* convergence.

a) B almost symmetric - local convergence (similar to AMG for SPD problems)

[H. De Sterck et al.]

b) B almost block diagonal (NCD) - global fast convergence

[P. Buchholz, T. Dayar, W. J. Stewart]

c) General B - several results

[I. Marek, P. Mayer, I. Pultarová]

AMG for SPD problems

Algebraic multigrid (AMG) for symmetric positive definite (SPD) problems

$$Ax = b$$

Let R and R^T are reduction and prolongation matrices.

The coarse problem $RAR^T z = Rr$, where r is the current residual.

The iteration matrix is $M = (I - T^m)(I - R^T(RAR^T)^{-1}RA)$, where

T is symmetric, commutes with A and $\rho(T) < 1$,

R is any matrix for which RAR^T is invertible, we have

$$\rho(M) < 1.$$

Advantageous choice of R: according to strongly connected elements.

Then rows of R represent low frequency vectors.

[A. Brandt, Algebraic multigrid theory: The symmetric case, 1983]

AMG for Helmholtz equation

Consider

$$Au = f,$$

a discretization of the indefinite Helmholtz equation

$$-\triangle u - k^2 u = f.$$

Fourier analysis:
eigenvalues of $I - \omega D^{-1}A$ (weighted Jacobi m.) are

$$\lambda_j = 1 - \omega\left(1 - \frac{\cos j\pi h}{1 - \frac{1}{2}k^2h^2}\right), \quad j = 1,\ldots,N, \quad h = \frac{1}{N}.$$

Fine grid - fast frequencies are eliminated from the error,
smoothed modes can be amplified !

Coarse grid - the problem becomes negative definite

Intermediate grid - several methods

E.g. [H. C. Elman, O. G. Ernst, D. P. O'Leary, 2001]

IAD for NCD problems

Nearly completely reducible (NCD) Markov chains.

Convergence estimate according to Stewart's book [1994].

[T. Dayar, W. J. Stewar, SIAM, 1996]:

"If B is a sum of a block diagonal matrix and E, where $\varepsilon := ||E||_2 \ll 1$

$$B = \begin{pmatrix} X & \varepsilon & \ldots & \varepsilon \\ \varepsilon & X & \ldots & \varepsilon \\ \ldots & \ldots & \ldots & \ldots \\ \varepsilon & \varepsilon & \ldots & X \end{pmatrix}$$

then the error is reduced by ε in every cycle of the Koury-McAllister-Stewart, Takahashi and Vantilborgh methods."

More precisely:

Sufficient conditions for global convergence with factor $O(\varepsilon)$ [W. L. Cao, W. J. Stewart, 1985]

1. B has block structure, for diagonal blocks $||B_{ii}||_2 = O(1)$ and $||B_{ij}||_2 = O(\varepsilon)$ for $i \neq j$, $i,j = 1,\ldots,p$.
2. There exists constant $M_1 > 0$ such that $||\hat{x}_i||_1 > M_1$, $i = 1,2,\ldots,p$.
3. Each block B_{ii} is similar to
$$\begin{pmatrix} 1 - O(\varepsilon) & 0 \\ 0 & H_i \end{pmatrix}$$
and there exists constant $M_2 > 0$ such that $||(I - H_i)^{-1}||_2 < M_2$ for $i = 1,2,\ldots,p$.
4. B is similar to
$$\begin{pmatrix} 1 - O(\varepsilon) & 0 \\ 0 & K \end{pmatrix}$$
and there exists constant $M_3 > 0$ such that $||(I - K)^{-1}||_2 < M_3\varepsilon^{-1}$.

But:
Hard to estimate.
For $||E||_2 \approx 1$ the method can diverge.

Two level IAD for general stochastic matrices

[J. Mandel, B. Sekerka, 1983; I. Marek, P. Mayer, 1998, 2003; U. Krieger, 1995]

Recall $B = P + Z$, $P = \hat{x}e^T$, aggregation groups, R and $S(y)$, e.g.

$$R = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad S(y) = \begin{pmatrix} 3/7 & 0 \\ 4/7 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{for} \quad y = \begin{pmatrix} 3/13 \\ 4/13 \\ 6/13 \end{pmatrix}.$$

Convention: elements in groups always consecutively numbered.

Error propagation matrices - derivation:

No eigenvalue of $RZS(y)$ is equal to one. *Proof:*

$$RZS(y)u = u$$

then $e^T u = 0$ then $PS(y)u = 0$ then

$$R(Z + P)S(y)u = u$$

and thus u is the Perron vector of an irreducible stochastic matrix $RBS(y)$ and thus $e^T u \neq 0$, which is a contradiction.

Having x^k and starting with the coarse problem $RBS(x^k)z = z$, $e^T z = 1$:

$$\begin{aligned} RBS(x^k)z &= z \\ R(Z+P)S(x^k)z &= z \\ RZS(x^k)z + R\hat{x}e^T S(x^k)z &= z \\ RZS(x^k)z + R\hat{x} &= z \\ R\hat{x} &= (I - RZS(x^k))z \\ (I - RZS(x^k))^{-1}R\hat{x} &= z. \end{aligned}$$

Then

$$\begin{aligned} S(x^k)z &= S(x^k)(I - RZS(x^k))^{-1}R\hat{x} \\ S(x^k)z &= (I - S(x^k)RZ)^{-1}S(x^k)R\hat{x} \\ S(x^k)z &= (I - P(x^k)Z)^{-1}P(x^k)\hat{x}, \end{aligned}$$

where $P(x^k) := S(x^k)R$ is a projection. Then

$$\begin{aligned} x^{k+1} - \hat{x} &= T^{\mu+\nu}(I - P(x^k)Z)^{-1}P(x^k)\hat{x} - \hat{x}, \\ x^{k+1} - \hat{x} &= T^{\mu+\nu}(I - P(x^k)Z)^{-1}P(x^k)\hat{x} - T^{\nu}(I - P(x^k)Z)^{-1}\hat{x}, \\ x^{k+1} - \hat{x} &= T^{\mu+\nu}(I - P(x^k)Z)^{-1}(P(x^k) - I)\hat{x}, \\ x^{k+1} - \hat{x} &= T^{\mu+\nu}(I - P(x^k)Z)^{-1}(I - P(x^k))(x^k - \hat{x}), \end{aligned}$$

Thus the error propagation matrix is

$$J(x^k) = T^{\mu+\nu}(I - P(x^k)Z)^{-1}(I - P(x^k)).$$

In case of convergence, the asymptotic error propagation matrix is

$$J(\hat{x}) = T^{\mu+\nu}(I - P(\hat{x})Z)^{-1}(I - P(\hat{x})).$$

Then

$$\rho(J(x^k)) < 1$$

would mean global convergence (it is not feasible!), and

$$\rho(J(\hat{x})) < 1$$

means local convergence.

Nevertheless, the Perron vector is a fixed point of all general IAD algorithms.
It is proved that the fixed point is unique - only for some special types of the IAD methods.

Some convergence criteria

Let $T = B$ and $\mu + \nu = 1$, so that $J(\hat{x}) = B(I - P(\hat{x})Z)^{-1}(I - P(\hat{x}))$, basic iteration by one step of power method.

Theorem. If at least one of the following options holds for B

1) one row is positive,
2) one column is positive;
3) the diagonal is positive;
4) stochastic complements of all diagonal blocks are primitive matrices,

then $\rho(J(\hat{x})) < 1$ (local convergence).

[1) Mandel, Sekerka, 1983; 2)-3) Marek, Pultarová, 2006; 4) Pultarová, 2008]

Counter-example.

$$B = \begin{pmatrix} 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 \\ 0 & 1 & 0 \end{pmatrix},$$

groups $G_1 = \{1\}$, $G_2 = \{2,3\}$. Divergence, $\rho(J(\hat{x})) = 1$.

Note that

- Only one step of basic iteration is allowed.
- Only $T = B$ is allowed. Power method.
- Only local convergence is obtained.

Thus not so much efficient, not robust.

Theorem. Let $T = B$ and $\mu + \nu = 1$. Let

a) $m < N$ and $G_1 = \{1\}, \ldots, G_m = \{m\}, G_{m+1} = \{m+1, \ldots, N\}$, and
b) the stochastic complement to $B_{G_{m+1}}$ be a primitive matrix.

Then $\rho(J(x^k)) < 1$ (global convergence).

[Ipsen, Kirkland, 2006].

Two levels, but more steps of basic iteration

Divergence in general:

Let C_{2n}, $N = 2n$, be a permutation (cyclic) matrix represented, for example for $n = 6$, by a directed path

$$1 \to 3 \to 5 \to 7 \to 9 \to 11 \to 12 \to 10 \to 8 \to 6 \to 4 \to 2 \to 1.$$

Theorem. Consider $B = C_{4n}$. Suppose $2n$ aggregation groups, each containing two elements. Let $T = C_{4n}$ and $\mu + \nu = n$. Then

$$\rho(J(\hat{x})) \geq n.$$

Theorem. Consider $B = C_{4n}$. Suppose $2n$ aggregation groups, each containing two elements. Let $\mu + \nu = n$ and let T correspond to the block-Jacobi iteration matrix with 2×2 blocks. Then

$$\rho(J(\hat{x})) \geq n.$$

More steps again, but better ordering

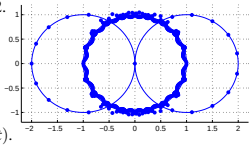
Ordering according to strong connections.

Let $\tilde{C}_N$ be defined by $[\tilde{C}_N]_{k+1,k} = 1$, $[\tilde{C}_N]_{1,N} = 1$ and $[\tilde{C}_N]_{i,j} = 0$ otherwise, e.g.

$$\tilde{C}_4 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Theorem. [P., 2009] For any N there exists a choice of the aggregation groups and $\mu + \nu$ such that $\limsup_{N\to\infty} \rho(J(\hat{x})) = 2$.

Example. $B = C_{600}$, $T^{\mu+\nu} = B^{N/2-1}$.
20 groups, each containing 30 elements.
Spectrum (thick dots) of the error matrix $J(\hat{x})$.
(Three thin circles help to recognize a location of the eigenvalues.)



Again, divergence in general, even in local sense.

Even better ordering (if possible) . . . fast convergence

Theorem. Let the block rows of the basic iteration matrix T be rank one matrices. Then the IAD method yields the exact solution after the second cycle.
[I. Marek, P. Mayer, 1998]

Example. Let

$$B = \begin{pmatrix} 1/5 & 0 & 1/5 \\ 4/5 & 0 & 4/5 \\ 0 & 1 & 0 \end{pmatrix},$$

groups $G_1 = \{1,2\}$, $G_2 = \{3\}$, let $T = B$.
The parts of x^1 are already parallel to the corresponding parts of $\hat{x}$. Then $x^2 = \hat{x}$.

Indeed, $\rho(J(\hat{x})) = 0$.

$$\hat{x} = \begin{pmatrix} 1/9 \\ 4/9 \\ 4/9 \end{pmatrix}.$$

Fourier analysis

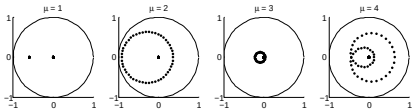
Fourier analysis enables quantitative estimates and optimal choice of parameters.
B must have some structure.

First results for B cyclic. $B = C_N$, where $[C_N]_{i+1,i} = 1$ and $[C_N]_{1,N} = 1$.

Theorem. [P, 2012] Let $N = 100$, $B = C_N$, (then $\hat{x} = e/N$). $T = \alpha B + (1-\alpha)I$, $\#G_i = 2$. Then spectrum of the error propagation matrix $J(\hat{x})$ is

$$\sigma(J(\hat{x})) = \{0, v_0, v_1, \ldots, v_{n-1}\},$$

where $v_k =$
$\frac{1}{2}\left(\left(1-e^{2\pi ki/N}\right)\left(1-\alpha+\alpha e^{-2\pi ki/N}\right)^{\mu} + \left(1+e^{2\pi ki/N}\right)\left(1-\alpha-\alpha e^{-2\pi ki/N}\right)^{\mu}\right)$.



Example. $T = \alpha B + (1-\alpha)I$, $\alpha = 0.8$ and $\mu + \nu \in \{1,2,3,4\}$. Spectra of $J(\hat{x})$. (The solid lines represent reference unit cycles.)

More than two levels, $L > 2$

Theorem. [P, 2012] Let $L = 3$, $\mu_m + \nu_m \geq 1$, $m = 1,2$. Let T commute with B.
The error in $n+1$-th cycle is

$$x^{n+1} - \hat{x} = J(x^n)(x^n - \hat{x}),$$

where

$$\begin{aligned} J(x^n) &= T^{\nu_1}\left((P_2T)^{\nu_2}(I-P_3Z)^{-1}\left((P_2-P_3)\sum_{k=0}^{\mu_2-1}(TP_2)^k(T-I)+I-P_3\right)\right. \\ &\quad \left. + \sum_{k=0}^{\nu_2-1}(P_2T)^k(I-P_2)\right)T^{\mu_1}, \end{aligned}$$

where

$$P_k = S(x^n)_1 S(x^n)_2 R_2 R_1$$

where R_k and $S(y)_k$ maps vectors from level k into level $k-1$ and vice versa, respectively.

Theorem. [P, 2012] Let $L \geq 2$ and $\mu_m = \nu_m = 1$ for all levels up to the coarsest one, $m = 1,2,\ldots,L-1$. Let T commute with B.
The error in $n+1$-th cycle is

$$x^{n+1} - \hat{x} = J(x^n)(x^n - \hat{x}),$$

where

$$\begin{aligned} J(x^n) &= T\prod_{k=2}^{L-1}(P_kT)(I-P_LZ)^{-1}\sum_{k=1}^{L-1}(P_k - P_{k+1})M_{k-1} \\ &\quad + T\sum_{m=1}^{L-2}\prod_{k=2}^{m}(P_kT)\sum_{k=1}^{m}(P_k - P_{k+1})M_{k-1}, \end{aligned}$$

where $M_0 = T$ and

$$M_k = \left(T + \sum_{j=2}^{k} TP_j(T-I)\right)T,$$

for $k = 1,2,\ldots,L-2$.

Some consequences

Previous two theorems $\Longrightarrow$ local convergence is not preserved

1. if we use $L+1$ levels instead of L levels;
2. if we use L levels instead of $L+1$ levels;
3. if we change μ_k and ν_k, but the sum $\mu_k + \nu_k$ remains the same.

Numerical example I. - Tandem queue

Serial connection of two servers.

Left: $\sigma(B)$, $N = 276$
$\lambda_2(B) = 1$
fourth largest eigenvalue 0.9890

Right: $\sigma(B_{new})$,
$B_{new} = \alpha B + (1-\alpha)I$
$\alpha = 0.7$
$\lambda_2(B_{new}) = 0.9923$

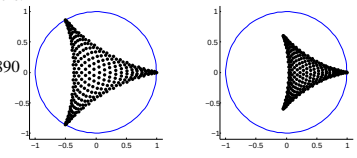


Table: Number of cycles and times for achieving the accuracy 10^{-6}.

N	$\#G_k$	power m.	cycles	time	IAD	cycles	time
36	2		-	-		24	0,02
	4		-	-		29	0,02
528	2		-	-		25	0,68
	4		-	-		30	0,38
2080	2		-	-		25	22,96
	4		-	-		28	10,78

Numerical example II. - Genetics

Matrix of transitions of genes (with a perturbation 10^{-5}), $N = 1200$.

Left: original B
Middle: reordered B
Right: $\sigma(B)$

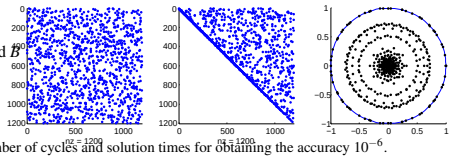
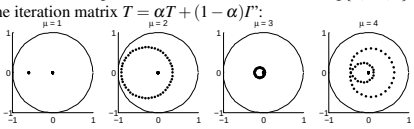


Table: Number of cycles and solution times for obtaining the accuracy 10^{-6}.

steps of basic it., $T = 0.8B + 0.2I$	1	2	**3**	4	5	6
time	8.2	28.5	**3.6**	8.2	3.7	4.9
cycles	25	83	**10**	22	9	12

Fourier analysis: "Three steps of basic iteration are best (among $\{1,\ldots,4\}$ for $\alpha = 0.8$ in the iteration matrix $T = \alpha T + (1-\alpha)I$":

Spectra of J's for B cyclic:



IAD metods

Definitions
Applications
Solution methods
Stationary
IAD
Convergence
(SPD, Helmholtz)
NCD
Two levels
More levels
Numerical examples

Numerical example II. - Genetics, cont.

Fourier analysis says:
In case of one step of basic iteration in every cycle, $\alpha = 1/2$ is best.
In case of two steps of basic iteration in every cycle, $\alpha = 1/3$ is best.

Table: **Two steps** of basic iteration. Number of cycles and times for achieving the accuracy 10^{-6}.

$\alpha =$	0.1	0.2	0.3	0.4	**0.5**	**0.6**	0.7	0.8	0.9
time	21.0	10.8	7.4	5.6	**4.3**	**3.6**	4.9	8.2	18.4
cycles	61	33	22	17	**13**	**11**	15	25	56

Table: **Three steps** of basic iteration. Number of cycles and times for achieving the accuracy 10^{-6}.

$\alpha =$	0.1	0.2	**0.3**	**0.4**	0.5	0.6	0.7	0.8	0.9
time	11.6	6.5.8	**4.4**	**3.8**	5.8	8.6	14.9	28.3	-
cycles	34	19	**13**	**11**	17	25	43	83	-

IAD metods

Definitions
Applications
Solution methods
Stationary
IAD
Convergence
(SPD, Helmholtz)
NCD
Two levels
More levels
Numerical examples

Recent papers and topics:

Adaptive smoothed aggregation multigrid for nonsymmetric problems (for Markov chains; with application to web ranking):
M. Brezina, H. De Sterck, T. A. Manteuffel, S. F. McCormick, K. Miller, Q. Nguyen, J. Pearson, J. Ruge, G. Sanders

Multilevel methods for Kronecker-based Markovian representations:
P. Buchholz, T. Dayar

Algebraic analysis of two-grid methods: The nonsymmetric case:
Y. Notay

Some open questions:

More criteria of convergence of the IAD methods, for more levels.

Special IAD methods for genes.

Fourier analysis of the IAD with larger groups, more levels, ...

Main question: Does local convergence always imply global convergence?

Stochastic Finite Element Methods (an "incomplete" introduction)

Bedřich Sousedík

University of Southern California, Los Angeles CA

parts are based on joint work with Roger G. Ghanem (USC) and Eric T. Phipps (Sandia)

SNA'13 conference, in honor of Professor Ivo Marek, January 2013

Essentially, all models are wrong, but some are useful.

George E. P. Box

Why stochastic models: uncertainty is everywhere

- Many applications (physical, biological, social, economic, etc.) are affected by a relatively large amount of **uncertainty**.
- As a result, **mathematical models of these processes should account for uncertainty**
- Accounting for uncertainty in processes governed by partial differential equations can involve
  - random coefficients,
  - random right-hand side (forcing terms),
  - random boundary conditions, initial conditions
  - random geometry, i.e., random boundary shapes

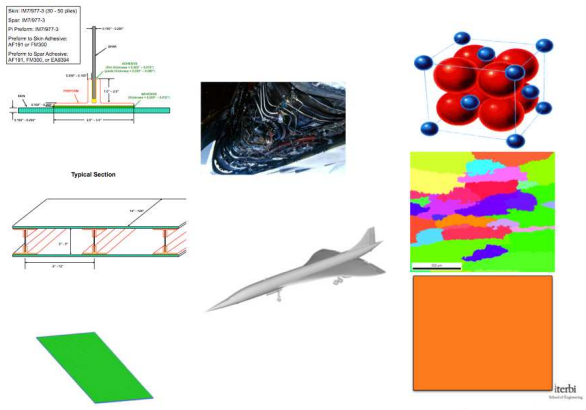
Examples

- forecasting financial markets (economic factors, human behavior, ...)
- modeling of wildfires (fuel, weather, ...),
- reliability of smart energy grids,
- development of renewable energy technologies,
- vulnerability analysis of water and power supplies,
- complex biological networks,
- climate change,
- design and licensing of nuclear reactors,
- etc.

Reasons for uncertainty

- available data are incomplete (incomplete description of parameters)
  - observable, but too difficult or costly to measure
    Example: media properties in oil reservoirs or aquifers
  - not observable/predictable
    Example: rainfall, wind shear
- not all scales in the data and/or solution can or should be resolved (there might be small, uresolved scales in the model that act as a kind of background noise ,i.e., macrobehavior from the microstructure).
  - it is too difficult (perhaps imposible) or costly to do so in a computational simulation
    Example: effect molecular scale (vibrations), turbulence
  - some scales may not be of interest
    Example: surface roughness, hourly stock prices

(image courtesy of Roger Ghanem)



Guidance in modeling

- Uncertainty is not a property of a system.
  It is a property of knowledge we have about that system.
- Knowledge evolves, and uncertainty should evolve accordingly.
  Question: evolution of vocabulary, or grammar, or both?
- If uncertainty reflects ignorance, then models of uncertainty should reflect on ignorance, on its sources and ways to manage it.

Types of uncertainties

Stochastic models give **quantitative** information about **uncertainty**.
In practice it is necessary to address the following types of uncertainties:

- Aleatoric - random, due to the intrinsic variability in the system
  Example: turbulent fluctuations of a flow field around an airplane wing, permeability in an aquifer, etc.
  → such variability is **inherent** and **irreducible**
- Epistemic - due to incomplete knowledge
  Example: mechanical properties of materials, etc.
  → can be **reduced** by experiments, improving measuring devices, etc.

Modelling noise (1)

- **White noise** - input data vary randomly and independently from one point of the domain to another and from one time instant to another
  - uncertainty is described in terms of uncorrelated random fields
  - Examples: surface roughness, porosity, thermal fluctuations
- **Colored noise** - input data vary randomly from one point of the physical domain to another and from one time instant to another according to a given (spatial/temporal) correlation structure
  - uncertainty is described in terms of correlated random fields
  - Examples: bone densities, rainfall amounts, permeabilities within subsurface layers

Modelling noise (2)

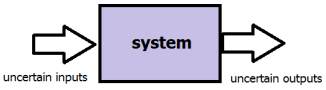
- **Random parameters** - input data depend on a finite[1] number of random parameters
  - each parameter may vary independently according to its own given probability density
  - alternately, the parameters may vary according to a given joint probability density

  Examples: homogeneous material properties, e.g, Young's modulus, Poisson's ratio, inflow mass, ...

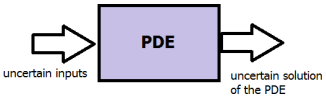
[1] What we really mean is that the number of parameters is not only finite, but independent of the spatial/temporal discretization; this is not possible for the approximation of white noise for which the number of parameters increases as the grid sizes decrease

Uncertainty Quantification (UQ)

Uncertainty Quantification (UQ) attempts to quantitatively assess the impact of input uncertainties on simulation outputs:



- of course, the system may have deterministic inputs as well.
We are interested in systems governed by partial differential equations:



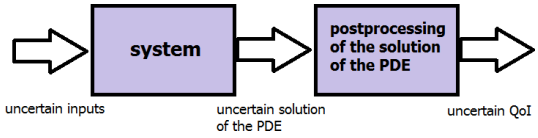
- the solution of the partial differential equation defines the mapping from the input variables to the output variables

Quantity of interest

Often, solutions of the PDEs are not the primary output

quantity of interest (QoI).

Quantities obtained by post-processing solutions of the PDE are often of interest



A realization of random system

A realization of the random system is determined by:

specifying a specific set of input variables

and then

using the PDE to determine the corresponding output variables.

- thus, a realization is a solution of a deterministic problem.

One is never interested in individual realizations of solutions of the PDE or of the quantities of interest.
- one is interested in determining statistical information about the quantities of interest, given statistical information about the inputs.

Suppose we have N random parameters $\{y_n\}_{n=1}^N$
- we use the abbreviation $\vec{y} = \{y_1, y_2, \ldots, y_N\}$
- each y_n could be distributed independently[2] according to its probability density function (pdf) $\rho_n(y_n)$ defined in a (possibly infinite) interval Γ_n.
- alternately, the parameters could be distributed according to a joint pdf $\rho(y_1, \ldots, y_N)$ that is a mapping from an $N-$dimensional set Γ into the real numbers
- independently distributed parameters are the special case for which

$$\rho(y_1, \ldots, y_N) = \prod_{n=1}^{N} \rho_n(y_n) \quad \text{and} \quad \Gamma = \Gamma_1 \otimes \Gamma_2 \otimes \cdots \otimes \Gamma_N$$

[2] Without proper justification and sometimes incorrectly, it is almost always assumed that the parameters are independent; based on empirical evidence, sometimes this is a justifiable assumption if the parameters are "knobs" case, but for correlated random fields, it is justifiable only for the (spherical) Gaussian case. In general, independence is a simplifying assumption that is involveed for the sake of convenience, e.g., because of a lack of knowledge.

- Realization = a solution $u(x, t, \vec{y})$ of a PDE for a specific choice $\vec{y} = \{y_n\}_{n=1}^N$ for the random parameters
  - again, there is no interest in individual realizations
- One may be interested in statistics of solutions of the PDE
  - average or expected value

$$\overline{u}(x, t) = \mathbb{E}[u(x, t; \cdot)] = \int_\Gamma u(x, t, \vec{y}) \rho(\vec{y})\, d\vec{y}$$

  - covariance

$$C_u(x, t; x', t') = \mathbb{E}\left[\left(u(x, t; \cdot) - \overline{u}(x, t)\right)\left(u(x', t'; \cdot) - \overline{u}(x', t')\right)\right]$$
$$= \int_\Gamma \left(u(x, t, \vec{y}) - \overline{u}(x, t)\right)\left(u(x', t', \vec{y}) - \overline{u}(x', t')\right) \rho(\vec{y})\, d\vec{y}$$

  - variance $C_u(x, t; x, t)$
  - higher moments

One may instead be interested in statistics of spatial/temporal integrals of the solution of the PDE
- for any fixed $\vec{y}$,

$$\mathcal{J}(t; \vec{y}) = \int_{\mathcal{D}} F(u; \vec{y}) dx \quad \text{or} \quad \mathcal{J}(x; \vec{y}) = \int_{t_0}^{t_1} F(u; \vec{y}) dt$$

  or

$$\mathcal{J}(\vec{y}) = \int_{t_0}^{t_1} \int_{\mathcal{D}} F(u; \vec{y})\, dx dt$$

  where $F(\cdot; \cdot)$ is given, $\mathcal{D}$ is a spatial domain, and (t_0, t_1) is a time interval.
- quantities defined with respect to integrals over boundary segments also often occur in practice

Examples (QoI): the space-time average of u,

$$\mathcal{J}(\vec{y}) = \int_{t_0}^{t_1} \int_{\mathcal{D}} u(x, t; \vec{y})\, dx dt$$

if u denotes a velocity field, then

$$\mathcal{J}(t; \vec{y}) = \int_{\mathcal{D}} u(x, t; \vec{y}) \cdot u(x, t; \vec{y})\, dx$$

is proportional to the kinetic energy

Again, one is not interested in the values of these quantities for specific choices of the parameters $\vec{y}$,

one is interested in their statistics.

Example: expected value of the kinetic energy

$$\mathbb{E}\left[\int_{\mathcal{D}} u(x, t; \vec{y}) \cdot u(x, t; \vec{y})\, dx\right]$$
$$= \int_\Gamma \int_{\mathcal{D}} u(x, t; \vec{y}) \cdot u(x, t; \vec{y}) \rho(\vec{y})\, dx\, d\vec{y}$$

Thus, quantities of interest of this common type involve integrals over the parameter space [3]

Example: for some $G(\cdot)$, integrals to the type

$$\int_\Gamma G\left(u(x, t; \vec{y})\right) \rho(\vec{y})\, d\vec{y} \quad \text{or possibly} \quad \int_\Gamma G\left(u(x, t; \vec{y}); x, t, \vec{y}\right) \rho(\vec{y})\, d\vec{y}$$

[3] An important class of quantities of interest that arises in, e.g., reliability studies, but that we do not have time to consider involves integrals over a subset of Γ; in particular, we have

$$\int_\Gamma \chi_{u_0} G(u(x; \vec{y})) \rho(\vec{y})\, d\vec{y} = \int_{\Gamma_{u_0}} G(u(x; \vec{y})) \rho(\vec{y})\, d\vec{y},$$

where, for some given u_0

$$\chi_{u_0} = \begin{cases} 1, & \text{if } u(x; \vec{y}) \geq u_0 \\ 0, & \text{otherwise} \end{cases} \quad \text{and} \quad \Gamma_{u_0} = \{\vec{y} \in \Gamma \text{ such that } u(x; \vec{y}) \geq u_0\}$$

Ideally, one wants to determine an approximation of the pdf for the quantity of interest, i.e.,

more than just a few statistical moments

of some output quantity

the quantity of interest is a pdf

one way (but not the only way) to construct the approximate pdf is to compute many statistical moments of the output quantity

so, again we are faced with evaluating stochastic integrals

Quadrature rules for stochastic integrals

Integrals of the type

$$\int_\Gamma G\left(u(x,t;\vec{y})\right)\rho(\vec{y})\,d\vec{y}$$

cannot, in general, be evaluated exactly.
Thus, these integrals are approximated using a quadrature rule

$$\int_\Gamma G\left(u(x,t;\vec{y})\right)\rho(\vec{y})\,d\vec{y} \approx \sum_{q=1}^{Q} w_q\, G\left(u(x,t;\vec{y}_q)\right)\rho(\vec{y}_q)$$

for some choice of

- quadrature weights $\{w_q\}_{q=1}^{Q}$ (real numbers)
- quadrature points $\{\vec{y}_q\}_{q=1}^{Q}$ (points in the parameter domain Γ)

Alternately, sometimes the probability density function is used in the determination of the quadrature points and weights so that instead one ends up with the approximation

$$\int_\Gamma G\left(u(x,t;\vec{y})\right)\rho(\vec{y})\,d\vec{y} \approx \sum_{q=1}^{Q} w_q\, G\left(u(x,t;\vec{y}_q)\right)$$

Monte-Carlo integration - the simplest rule $\Longrightarrow$

- randomly select Q points in Γ according to the pdf $\rho(\vec{y})$
- evaluate the integrand at each of the sample points
- average the values so obtained , i.e., for all q, $w_q = 1/Q$

Big problem

In practice, one usually does not know much about the statistics of the input variables

- one is lucky if one knows a range of values, e.g., maximum or minimum values, for an input parameter (in which case one after assumes that the parameter is uniformly distributed over that range),
- if one is luckier, one knows the mean and variance for the input parameter (in which case one often assumes that the parameter is normally distributed),
- of course, one may be completely wrong in assuming such simple probability distributions for a parameter.

This leads to the need to solve stochastic model calibration problem.

Model calibration

Model calibration is the task of determining statistical information about the inputs of a system, given statistical information about the outputs

- e.g., one can use experimental observations to determine the statistical information about the outputs
- in particular, one wants to identify the probability density function (pdf) of the input variables

Of course, the system still maps the input to the outputs

- thus, determining the input pdf is an inverse problem
- usually involves an iteration in which guesses for the input pdf are updated
- several ways to do the update, e.g., Bayesian, maximum likelihood, ...

Colored noise: correlated random fields

- We now consider correlated random fields $\eta(x,t;\omega)$
  - at each point x in a spatial domain $\overline{\mathcal{D}}$ and at each instant t in a time interval $[t_0,t_1]$, the value of η is determined by a random variable ω whose values are drawn from a given probability distribution
  - however, unlike the white noise case, the covariance function of the random field $\eta(x,t;\omega)$ does not reduce to delta function
- In rare cases, a formula for the random field is "known"
  - again, we cannot sample the random field at every spatial and temporal point
  - on the other hand, unlike the white noise case, the fact that the random field is correlated implies that one can find a discrete approximation to the random field for which the number of degrees of freedom can be thought of as fixed, i.e., independent of the spatial and temporal grid sizes

- More often, only the
  mean $\mu_\eta(x,t)$
  and
  covariance function $\mathrm{cov}_\eta(x,t;x',t'0)$
  are known for points x, x' in $\overline{\mathcal{D}}$ and time instants t, t' in $[t_0,t_1]$
  - in this case, we do not have a formula for $\eta(x,t;\omega)$
  - thus, we cannot evaluate $\eta(x,t;\omega)$ when we need to
  - for example, if $\eta(x,t;\omega)$ is a coefficient or a forcing function in a PDE, then to determine an approximate realization of the PDE we need to evaluate $\eta(x,t;\omega)$ for a specific choice of ω and at specific points x and specific instants of time t used in the discretized PDE

- Examples of covariance functions

$$\mathrm{cov}(x,t;x',t') = e^{-|x-x'|/L-|t-t'|/T}$$

and

$$\mathrm{cov}(x,t;x',t') = e^{-|x-x'|^2/L^2-|t-t'|^2/T^2}$$

where L is the correlation length and T is the correlation time
  - large L, $T \Longrightarrow$ long-range order
  - small L, $T \Longrightarrow$ short-range order
- Note that the covariance functions are symmetric and positive

So, we have two cases:
- (more common:) only the mean and covariance are known
  - we would like to find a simple formula depending on only a few parameters whose mean and covariance function are approximately the same as the given mean and covariance function
- (rare:) random field is given as a formula, but we want to approximate it
  - we would like to approximate it using few random parameters, certainly with a number of parameters that is independent of the spatial and temporal grid sizes
  - of course, this case can be turned into the first case by determining the mean and covariance function of the given random field (this may or may not be a good idea)

Among the known ways for doing these tasks, we will focus on perhaps the most popular

the Karhunen–Loève (KL) expansion of a random field $\eta(x,t;\omega)$

Given the mean and covariance of a random field $\eta(x,t;\omega)$
- the KL expansion provides a simple formula that can be used whenever one needs a value $\eta(x,t;\omega)$

- to keep things simple, we discuss KL expansions for the case of spatially-dependent random fields

The Karhunen–Loève expansion

Given the mean $\mu_\eta(x)$ and covariance $\mathrm{cov}_\eta(x,x')$ of a random field $\eta(x,\omega)$, determine the eigenpairs $\{\lambda_n, v_n(x)\}_{n=1}^\infty$ from the eigenvalue problem

$$\int_{\mathcal{D}} \mathrm{cov}_\eta(x,x')\, v(x')\, dx' = \lambda v(x)$$

- often in practice, an approximate version of this problem is solved, e.g., using a finite element method
- due to the symmetry of $\mathrm{cov}_\eta(\cdot,\cdot)$, the eigenvalues λ_n are real and the eigenfunctions $v_n(x)$ can be chosen to be real and orthonormal, i.e.,

$$\int_{\mathcal{D}} v_n(x) v_m(x)\, dx = \delta_{mn}$$

- due to positivity of $\eta(x;\omega)$, the eigenvalues are all positive
  - without loss of generality, they may be ordered in non-increasing order $\lambda_1 \geq \lambda_2 \geq \ldots$

Then, the random field $\eta(x;\omega)$ admits the KL expansion[4]

$$\eta(x;\omega) = \mu_\eta(x) + \sum_{n=1}^{\infty} \sqrt{\lambda_n} v_n(x) Y_n(\omega),$$

where $\{Y_n(\omega)\}_{n=1}^\infty$ are centered and uncorrelated random variables, i.e.,

$$\mathbb{E}(Y_n(\omega)) = 0 \quad \mathbb{E}(Y_n(\omega) Y_m(\omega)) = 0$$

that inherit the probability structure of the random field $\eta(x;\omega)$
- e.g., if $\eta(x;\omega)$ is a Gaussian random field, then the Y_n's are all Gaussian random variables

[4] Let's see the next slide.

To see this, let's suppose

$$\eta(x;\omega) = \mu_\eta(x) + \sum_{n=1}^{\infty} a_n b_n(x) y_n(\omega)$$

where

$$\int_{\mathcal{D}} b_n(x) b_{n'}(x)\, dx = \delta_{nn'}, \quad \mathbb{E}(y_n) = 0, \quad \mathbb{E}(y_n y_{n'}) = \delta_{nn'}$$

i.e., $\{b_n(\cdot)\}_{n=1}^{\infty}$ is a set of orthonormal functions and $\{y_n(\cdot)\}_{n=1}^{\infty}$ is a set of uncorrelated random variables; we then have that

$$\mathbb{E}\left(\left(\eta(x;\cdot) - \mu_\eta(x)\right)\left(\eta(x';\cdot) - \mu_\eta(x')\right)\right) = \sum_{n=1}^{\infty}\sum_{n'=1}^{\infty} a_n a_{n'} b_n(x) b_{n'}(x') \mathbb{E}(y_n y_{n'}) = \sum_{n=1}^{\infty} a_n^2 b_n(x) b_n(x')$$

so that

$$\mathrm{cov}_\eta(x,x') = \sum_{n=1}^{\infty} a_n^2 b_n(x) b_n(x')$$

then, we have that

$$\int_{\mathcal{D}} \mathrm{cov}_\eta(x,x') b_{n'}(x') dx' = \sum_{n=1}^{\infty} a_n^2 b_n(x) \int_{\mathcal{D}} b_n(x') b_{n'}(x') dx' = a_{n'}^2 n_{n'}(x)$$

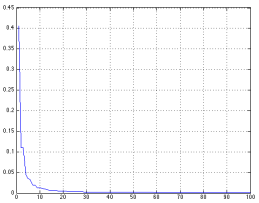
so that indeed $\{a_n^2, b_n(x)\}_{n=1}^{\infty}$ are the eigenpairs, i.e., we recover the KL expansion

The usefulness of the KL expansion results from the fact that

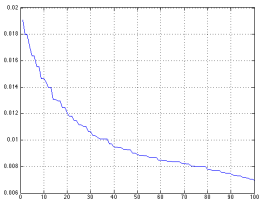
the eigenvalues $\{\lambda_n\}_{n=1}^{\infty}$ decay as n increases

- how fast they decay depends on the smoothness of the covariance function $\mathrm{cov}_\eta(x,x')$ and on the correlation length L

Eigenvalue decay vs. correlation length



$L = 0.5$



$L = 0.05$

$$\mathrm{cov}(x,x') = \sigma^2 e^{-|x-x'|^2/L^2}$$

The decay of the eigenvalues implies that truncated KL expansions

$$\eta_N(x;\omega) = \mu(x) + \sum_{n=1}^{N} \sqrt{\lambda_n} v_n(x) Y_n(\omega)$$

can be accurate approximations to the exact expansion

- if one wishes for the relative error to be less than a prescribed tolerance δ, i.e., if one wants

$$\frac{\|\eta_N - \eta\|^2}{\|\eta\|^2} \leq \delta,$$

one should choose N to be the smallest integer such that

$$\frac{\sum_{n=N+1}^{\infty} \lambda_n}{\sum_{n=1}^{\infty} \lambda_n} \leq \delta \quad \text{or, equivalently,} \quad \frac{\sum_{n=1}^{N} \lambda_n}{\sum_{n=1}^{\infty} \lambda_n} \geq 1 - \delta$$

- Although the Y_n's are uncorrelated, in general they are not independent
- in fact, they are independent if and only if they are (spherical) Gaussian
- however, every random field can, in principle, be written as a function of a Gaussian random field
  - the inverse of the cumulative probability density of the given field

  so that, in this way, we only have to deal with Gaussian random variables
- Dealing with independent random variables can have important practical consequences

One important issue is the well posedness of the PDE when using a KL representation of random fields

- suppose the coefficient $a(x;\omega)$ of an elliptic PDE is a random field
  - it cannot be a Gaussian random field since then it would admit negative values, which is not allowable
- one way to get around this is to let, with $a_{min} > 0$,

$$a(x;\omega) = a_{min} + e^{\eta(x;\omega)}$$

  where $\eta(x;\omega)$ is a Gaussian random field with given mean and covariance
- then, using a truncated KL expansion for $\eta(x;\omega)$ we have that

$$a(x;\omega) = a_{min} + e^{\mu(x) + \sum_{n=1}^{N} \sqrt{\lambda_n} v_n(x) Y_n(\omega)}$$

  where $\{Y_n(\omega)\}_{n=1}^{N}$ are Gaussian random variables

PDE's with random inputs depending on random parameters

One or more
input functions,
e.g., coefficients, forcing terms, initial data, etc. in a PDE
depend on a finite number of random parameters,

- the input function could also depend on space and time
- the random parameters could come from a Karhunen-Loève expansion of a correlated random field
- the random parameters could appear naturally in the definition of input function, e.g., the Young's modulus or a diffusivity coefficient could be random

Ideally, we would know the probability density function (PDF) for each parameter

- as has already been mentioned, in practice, we know very little about the statistics of input parameters
- however, we will assume that we know the PDFs for all the random input parameters

The well-posedness of the PDE for all possible values of the parameters is a very important (and sometimes ignored) consideration

- for the simple elliptic PDE

$$\nabla \cdot (a(x; y_1, \ldots, y_N) \nabla u) = f(x), \quad \text{in } \mathcal{D}$$

we must have, for some $0 < a_{min} \leq a_{max}$,

$$a_{min} \leq a(x; y_1, \ldots, y_N) \leq a_{max} \quad \forall x \in \mathcal{D} \text{ and } \forall \vec{y} \in \Gamma$$

- this could place a constraint on how one chooses the PDF for the parameters
- for example, if we have

$$a(x; y) = a_0 + y$$

where $a_0 > 0$, we cannot choose y to be Gaussian random parameter

A brief overview of numerical methods for stochastic PDEs

Stochastic finite element methods (SFEMs)
$\Longrightarrow$ methods for which spatial discretization is effected using finite element methods

- Stochastic Galerkin methods (SGMs)
  $\Longrightarrow$ methods for which probabilistic discretization is also effected using a Galerkin method
  - polynomial chaos and generalized polynomial chaos are SGMs
  - we will also consider other SGMs
- Stochastic sampling methods (SSMs)
  $\Longrightarrow$ points in the parameter domain Γ are sampled, then used as inputs for PDE, and then ensemble averages of output quantities of interest are computed
  - Monte-Carlo finite element methods are the simplest SSMs
  - stochastic collocation methods (SCMs) are also SSMs
    - the sampling points are the quadrature points corresponding to some quadrature rule

Spaces used in numerical methods for stochastic PDEs

- Let $\mathcal{D} \in \mathbb{R}^d$ denote a spatial domain with boundary $\partial\mathcal{D}$
  - $d = 1, 2, 3$ denotes the spatial dimension
  - $x \in \mathcal{D}$ denotes the spatial variable
- Let $\Gamma \in \mathbb{R}^N$ denote a parameter domain
  - N denotes the number of parameters
  - $\vec{y} = (y_1, \ldots, y_N) \in \Gamma$ denotes the random parameter vector
  - note that we have a finite number of parameters $\{y_n\}_{n=1}^N$ but they can take on values anywhere in the Euclidean domain Γ.

Let $u(x; \vec{y}) \in X \times Z$ denote the solution of the SPDE
- often X is the Sobolev space such as $H_0^1(\mathcal{D})$
- generally $Z = L_\rho^q(\Gamma)$ is the space of functions of N variables whose $q-$th power is integrable wrt the joint PDF (the weight fction) $\rho(\cdot)$, i.e., those functions $g(\vec{y})$ for which

$$\int_\Gamma |g(\vec{y})|^q \, \rho(\vec{y}) \, d\vec{y} < \infty$$

- q is chosen according to how many statistical moments one wants to have well defined
- the most common choice is $q = 2$ so that up to the second moments are well defined
- if $\{y_1, \ldots, y_N\}$ are independent and if $L_{\rho_n}^q(\Gamma_n)$ denotes the space of functions that have integrable $q-$th powers wrt the PDF $\rho_n(y_n)$, we have that

$$L_\rho^q(\Gamma) = L_{\rho_1}^q(\Gamma_1) \otimes L_{\rho_2}^q(\Gamma_2) \otimes \cdots \otimes L_{\rho_N}^q(\Gamma_N)$$

- It is entirely natural to treat a function $u(x;\vec{y})$ of d spatial variables and of N random parameters as a function of $d+N$ variables
- We will assume here that all methods use the same approach to effect discretization wrt the spatial variables
  (we focus on finite element methods → stochastic FEM)
- We assume that Γ is a parameter box
  - without loss of generality, Γ can be taken to be a hypercube in $\mathbb{R}^N$
  - for parameters with unbounded PDFs, Γ can be of infinite extent
  - if the parameters are constrained, Γ need not be so simple, e.g., if y_1 and y_2 are independent except that we require that $y_1^2 + y_2^2 \leq 1$, then Γ would be the unit circle

Representation of random variables using polynomial chaos

Probability spaces and random variables

- Let $(\Omega, \mathcal{F}, \mathcal{P})$ be a probability space
  - Ω is an event space
  - $\mathcal{F}$ is a $\sigma-$algebra on Ω
  - $\mathcal{P}$ is a probability measure
- Random variables are functions $X : \Omega \to \mathbb{R}$ with a measure corresponding to their image:
  - if $X^{-1}(A) \in \mathcal{F}$, then define $\mu(A) = \mathcal{P}(X^{-1}(A))$.
  - $p(x) = d\mu/dx$; the density of the random variable X (with respect to the Lebesgue measure on $\mathbb{R}$).
  - Expectation: $\langle f \rangle = \int f \, d\mu = \int f \, p(x) \, dx$
- Let $\xi : \Omega \to \mathbb{R}^N$ such that for $i = 1, \ldots, N$ each $\xi_i : \Omega \to \mathbb{R}$, be a set of random variables
- $\mathcal{F}(\xi)$: $\sigma-$algebra generated by the set ξ of random variables
- $L^2(\Omega, \mathcal{F}(\xi), \mathcal{P})$: Hilbert space of real-valued random variables defined on $(\Omega, \mathcal{F}(\xi), \mathcal{P})$ with finite second moments

One dimensional polynomial chaos expansion

Consider:

$$u = \sum_{k=0}^{P} u_k \psi_k(\xi)$$

- u: random variable (RV) represented with 1D PCE
- u_k: PC coefficients (deterministic)
- ψ_k: 1D Hermite polynomial of order k
- ξ: Gaussian RV

A random quantity is represented with an expansion consisting of functions of random variable multiplied with deterministic coefficients

- Set of deterministic PC coefficients fully describes RV
- Separates randomness from deterministic dimensions

One-dimensional Hermite polynomials

(Note: this is probabilistics', not physicists', definition)

$$\begin{aligned} \psi_0(\xi) &= 1, \\ \psi_k(\xi) &= (-1)^k e^{\xi^2/2} \frac{d^k}{d\xi^k} e^{-\xi^2/2}, \quad k = 1, 2, \ldots \\ \psi_1(\xi) &= \xi, \quad \psi_2(\xi) = \xi^2 - 1, \quad \psi_3(\xi) = \xi^3 - 3\xi, \quad \ldots \end{aligned}$$

The Hermite polynomials form an orthogonal basis over $[-\infty, \infty]$ with respect to the inner product

$$\langle \psi_i, \psi_j \rangle = \int_{-\infty}^{\infty} \psi_i(\xi)\psi_j(\xi) w(\xi) \, d\xi = \delta_{ij} \langle \psi_i^2 \rangle$$

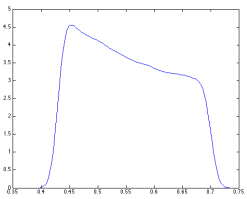
where $w(x)$ is the weight function

$$w(\xi) = \frac{1}{\sqrt{2\pi}} e^{-\xi^2/2}.$$

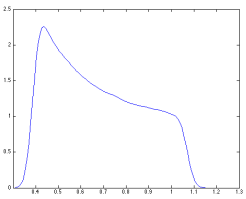
Note that $w(\xi)$ is the density of a standard normal random variable.

Example of one-dimensional polynomials: pdf plots

$$u = 0.5 + 0.2\phi_1(\xi) + 0.1\phi_2(\xi)$$



Hermite pol.



Legendre pol.

Multidimensional Hermite polynomials

The multidimensional Hermite polynomial $\Psi_i(\xi_1, \ldots, \xi_n)$ is a tensor product of the 1D Hermite polynomials, with a suitable multi-index

$$\alpha^i = (\alpha_1^i, \alpha_2^i, \ldots, \alpha_n^i),$$

and

$$\Psi_i(\xi_1, \ldots, \xi_n) = \prod_{k=1}^{n} \psi_{\alpha_k^i}(\xi_k)\,.$$

For example, 2D Hermite polynomials:

i	p	Ψ_i
0	0	1
1	1	ξ_1
2	1	ξ_2
3	2	$\xi_1^2 - 1$
4	2	$\xi_1\xi_2$
5	2	$\xi_2^2 - 1$
...	...	...

Multidimensional inner products – orhogonality

$$\begin{aligned}\langle \Psi_i \Psi_j \rangle &\equiv \int \ldots \int \Psi_i(\xi)\Psi_j(\xi) w(\xi_1) w(\xi_2) \ldots w(\xi_n)\, d\xi_1 d\xi_2 \ldots d\xi_n \\ &= \prod_{k=1}^{n} \left\langle \psi_{\alpha_k^i}(\xi_k)\psi_{\alpha_k^j}(\xi_k) \right\rangle = \delta_{ij} \left\langle \Psi_i^2 \right\rangle\end{aligned}$$

where, $w(\xi) = \frac{1}{\sqrt{2\pi}} e^{-\xi^2}/2$

such that

$$\begin{aligned} u = \sum_{k=0}^{P} u_k \Psi_k \quad &\Rightarrow \quad \langle \Psi_i u \rangle = \sum_{k=0}^{P} u_k \langle \Psi_i \Psi_k \rangle = u_i \langle \Psi_i^2 \rangle \\ &\Rightarrow \quad u_i = \frac{\langle u \Psi_i \rangle}{\langle \Psi_i^2 \rangle} \end{aligned}$$

Multidimensional polynomial chaos expansion (1)

Consider:

$$u(\omega) = \sum_{k=0}^{P} u_k \Psi_k(\xi_1(\omega), \ldots, \xi_n(\omega))$$

- u: Random Variable (RV) represented with multidimensional PCE
- u_k: PC coefficients (deterministic)
- Ψ_k: Multidimensional Hermite polynomials up to order p
- ξ_i: Gaussian RV
- n: Dimensionality of stochastic space
- $P+1$: Number of PC terms: $P + 1 = \frac{(n+p)!}{n!p!}$

The number of stochastic dimensions represents the number of independent inputs, degrees of freedom that affect the random variable u

- E.g., one stochastic dimension per uncertain model parameter
- Contributions from each uncertain input can be identified
- Compact representation of random variable and its dependencies

Multidimensional polynomial chaos expansion (2)

A r.v. $u(\omega)$ in $L^2(\Omega, \mathcal{F}(\xi), \mathcal{P})$ can be described by a PC expansion in terms of: the infinite-dimensional i.i.d. Gaussian basis $\xi = \{\xi_i(\omega)\}_{i=1}^{\infty}$

$$\begin{aligned} u(\omega) &= a_0 \Gamma_0 + \sum_{i_1=1}^{\infty} a_{i_1} \Gamma_1(\xi_{i_1}(\omega)) \\ &+ \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{\infty} a_{i_1 i_2} \Gamma_2(\xi_{i_1}(\omega), \xi_{i_2}(\omega)) \\ &+ \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{\infty}\sum_{i_3=1}^{\infty} a_{i_1 i_2 i_3} \Gamma_3(\xi_{i_1}(\omega), \xi_{i_2}(\omega), \xi_{i_3}(\omega)) + \ldots \end{aligned}$$

where Γ_p is the Polynomial Chaos of order p, $\Gamma_0 = 1$, and

$$\Gamma_p(\xi_{i_1}, \ldots, \xi_{i_p}) = (-1)^p e^{\frac{1}{2}\xi^T\xi} \frac{\partial^p}{\partial\xi_{i_1} \ldots \partial\xi_{i_p}} e^{-\frac{1}{2}\xi^T\xi}$$

Notes on the PC construction

- The Polynomial Chaoses are by construction orthogonal with respect to the Gaussian probability measure
- They are thus identical with the corresponding multidimensional Hermite polynomials
- The first four PCs are given by

$$\begin{aligned} \Gamma_0 &= 1 \\ \Gamma_1(\xi_{i_1}) &= \xi_{i_1} \\ \Gamma_2(\xi_{i_1}, \xi_{i_2}) &= \xi_{i_1}\xi_{i_2} - \delta_{i_1 i_2} \\ \Gamma_3(\xi_{i_1}, \xi_{i_2}, \xi_{i_3}) &= \xi_{i_1}\xi_{i_2}\xi_{i_3} - \xi_{i_1}\delta_{i_2 i_3} - \xi_{i_2}\delta_{i_1 i_3} - \xi_{i_3}\delta_{i_1 i_2} \\ &\ldots \end{aligned}$$

[R. G. Ghanem, and P. D. Spanos, 1991]

A more compact notation

- An L^2 random variable $u(x, t, \omega)$ can be described by a PC expansion in terms of:
  - Hermite polynomials Ψ_k, $k = 1, \ldots, \infty$
  - the associated infinite-dimensional Gaussian basis $\{\xi_i(\omega)\}_{i=1}^{\infty}$
  - spectral mode strengths $u_k(x, t)$, $k = 1, \ldots, \infty$
- Truncated to finite dimension n and order p, the PC expansion for u is written as

$$u(x, t, \omega) \simeq \sum_{k=0}^{P} u_k(x, t) \Psi_k(\xi(\omega))$$

where $\xi(\omega) = \{\xi_1(\omega), \ldots, \xi_n(\omega)\}$, and $P + 1 = \frac{(n+p)!}{n!p!}$

Generalized Polynomial Chaos (gPC)

PC type	Domain	Density $w(\xi)$	Polynomial	Free parameters
Gauss	$(-\infty, \infty)$	$\frac{1}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}}$	Hermite	none
uniform	$[-1, 1]$	$\frac{1}{2}$	Legendre	none
Gamma	$[0, \infty)$	$\frac{x^\alpha e^{-\xi}}{\Gamma(\alpha+1)}$	Laguerre	$\alpha > -1$
Beta	$[-1, 1]$	$\frac{(1+\xi)^\alpha (1-\xi)^\beta}{2^{\alpha+\beta+1} B(\alpha+1, \beta+1)}$	Jacobi	$\alpha > -1, \beta > -1$

Inner product: $\langle \psi_i \psi_j \rangle \equiv \int_a^b \psi_i(\xi) \psi_j(\xi) w(\xi)\, d\xi$

- Wiener-Askey scheme provides a hierarchy of possible continuous PC bases, see Xiu and Karniadakis, SISC, 2002.
- Input parameter domain often dictates the most convenient choice of PC
- Polynomial functions can also be tailored to be orthogonal w.r.t. chosen, arbitrary, density

Postprocessing/Analysis

- Moments
- Plotting PDFs of RVs represented with PCEs
- When is a PCE accurate enough?

Moments of RVs described with PCEs

$$u = \sum_{k=0}^{P} u_k \Psi_k(\xi)$$

- Expectation: $\langle u \rangle = u_0$
- Variance: σ^2

$$\begin{aligned}
\sigma^2 &= \left\langle (u - \langle u \rangle)^2 \right\rangle \\
&= \left\langle \left(\sum_{k=1}^{P} u_k \Psi_k(\xi) \right)^2 \right\rangle \\
&= \left\langle \sum_{k=1}^{P} \sum_{j=1}^{P} u_j u_k \Psi_j(\xi) \Psi_k(\xi) \right\rangle \\
&= \sum_{k=1}^{P} \sum_{j=1}^{P} u_j u_k \left\langle \Psi_j(\xi) \Psi_k(\xi) \right\rangle = \sum_{k=1}^{P} u_k^2 \left\langle \Psi_k(\xi)^2 \right\rangle
\end{aligned}$$

Plotting PDFs of RVs correponding to PCEs

$$u = \sum_{k=0}^{P} u_k \Psi_k(\xi)$$

- Analytical formula for PDF(u) exists
  - Involves polynomial root finding, and is hard to generalize to multi. dim.
- PCE is cheap to sample
  - Brute-force sampling and bin samples into histogram
  - Use Kernel Density Estimation (KDE) to get smoother PDF with fewer samples u_i

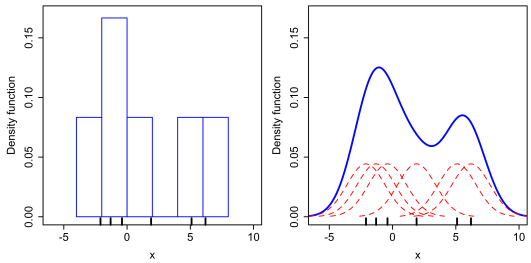
$$PDF(u) = \frac{1}{N_s h} \sum_{i=1}^{N_s} K\left(\frac{u - u_i}{h} \right)$$

K is the kernel, h is the bandwidth

Comparison of histograms and KDE

KDE ... Kernel Density Estimation

(source: Wikipedia)



- Bandwidth h needs to be chosen carefully to avoid over smoothing

How do I know my PCE has converged?

- Approximation error in PCE is topic of a lot of research
- Often, rules of thumb:
  - Higher order PC coefficients should decay
  - Increase order until results no longer change
  - Not always fail-proof, ...

Propagation of uncertain inputs represented with PCEs

- Collocation approaches
  - Non-intrusive: Match PCE to random variable at chosen sample points
- Galerkin projection approaches project uncertain quantity onto space covered by PC basis functions
  - Relying on orthogonality of basis functions

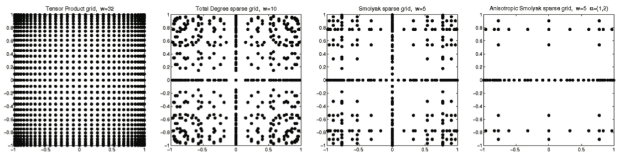
$$u_k = \frac{\langle u\Psi_k \rangle}{\langle \Psi_k^2 \rangle}. \quad k = 0, \ldots, P$$

  - Intrusive: project governing equations
    - Residual orthogonal to space of basis functions

Non-intrusive spectral stochastic UQ formulation

Let $\Theta_M = \{\xi^{(j)}\}_{j=1}^M \subset \Gamma$ be a set of prescribed nodes in the random space, where $\Theta_M = \theta_1^1 \times \cdots \times \theta_1^N$ and M is the total number of nodes

(reproduced from Babuška et al., SIAM Review (2010))



Solve a governing (deterministic) equation for each $u(x;\xi^{(j)})$ and interpolate

$$u(x;\xi) = \sum_{j=1}^{M} u(x;\xi^{(j)}) L_j(\xi)$$

Intrusive spectral stochastic UQ formulation

- Sample ODE with parameter λ:

$$\frac{du}{dt} = \lambda u$$

- Let λ be uncertain; introduce $\xi \sim \mathcal{N}(0,1)$.
- Express λ and u using PCEs in ξ:

$$\lambda = \sum_{k=0}^{P} \lambda_k \Psi_k(\xi), \quad u(t) = \sum_{k=0}^{P} u_k(t) \Psi_k(\xi)$$

- Substitute in ODE and apply a Galerkin projection on $\Psi_i(\xi)$,

Galerkin projection on $\Psi_i(\xi)$

$$\begin{aligned}
\frac{d}{dt}\left(\sum_{k=0}^{P} u_k(t)\Psi_k(\xi)\right) &= \left(\sum_{p=0}^{P} \lambda_p \Psi_p(\xi)\right)\left(\sum_{q=0}^{P} u_q(t)\Psi_q(\xi)\right) \\
\sum_{k=0}^{P} \frac{du_k(t)}{dt}\Psi_k(\xi) &= \sum_{p=0}^{P}\sum_{q=0}^{P} \lambda_p u_q(t) \Psi_p(\xi)\Psi_q(\xi) \\
\left\langle \sum_{k=0}^{P} \frac{du_k(t)}{dt}\Psi_k(\xi)\Psi_i(\xi)\right\rangle &= \left\langle \sum_{p=0}^{P}\sum_{q=0}^{P} \lambda_p u_q(t) \Psi_p(\xi)\Psi_q(\xi)\Psi_i(\xi)\right\rangle \\
\sum_{k=0}^{P} \frac{du_k(t)}{dt}\langle \Psi_k(\xi)\Psi_i(\xi)\rangle &= \sum_{p=0}^{P}\sum_{q=0}^{P} \lambda_p u_q(t) \langle \Psi_p(\xi)\Psi_q(\xi)\Psi_i(\xi)\rangle \\
\frac{du_k(t)}{dt}\langle \Psi_i^2\rangle &= \sum_{p=0}^{P}\sum_{q=0}^{P} \lambda_p u_q(t) \langle \Psi_p \Psi_q \Psi_i\rangle
\end{aligned}$$

Resulting spectral ODE system

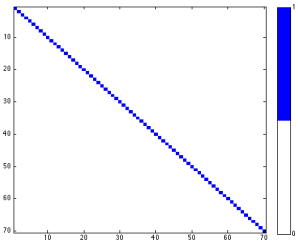
- $(P+1)$-dimensional ODE system

$$\frac{du_k}{dt} = \sum_{i=0}^{P}\sum_{j=0}^{P} c_{ijk}\lambda_i u_j, \qquad k = 0, \ldots, P,$$

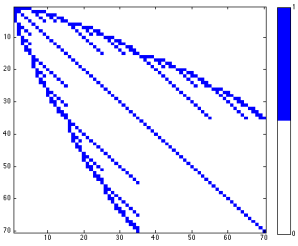
  where $c_{ijk} = \langle \Psi_i \Psi_j \Psi_k \rangle / \langle \Psi_k^2 \rangle$
- The tensor c_{ijk} can be evaluated once and stored for any given PC order and dimension
- This tensor is sparse, i.e., many elements are zero

Example of c_{ijk}

$n = 4$ ... stochastic dimension
$p = 4$ ... order of the polynomial expansion of u
p_λ ... order of the polynomial expansion of the coefficient λ



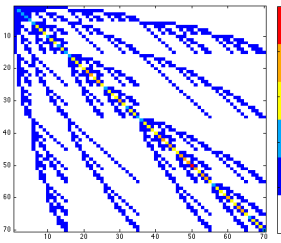
$p_\lambda = 0$



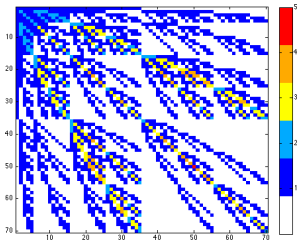
$p_\lambda = 1$

Example of c_{ijk}

$n = 4$ ... stochastic dimension
$p = 4$ ... order of the polynomial expansion of u
p_λ ... order of the polynomial expansion of the coefficient λ



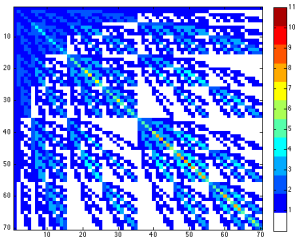
$p_\lambda = 2$



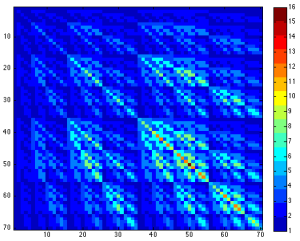
$p_\lambda = 3$

Example of c_{ijk}

$n = 4$ ... stochastic dimension
$p = 4$ ... order of the polynomial expansion of u
p_λ ... order of the polynomial expansion of the coefficient λ



$p_\lambda = 4$



$p_\lambda = 8$

Challenges for PCE-based Uncertainty Quantification

- Representing input variables with arbitrary distributions
- Systems with high-dimensional uncertainty
- Systems with long time horizon / oscillatory behavior
- Nonlinearities in governing equations for intrusive UQ
- Physical constraints in uncertain quantities
- Systems with non-smooth behavior - discontinuities
- Systems with inherent stochasticity

Various approaches have been developed to tackle these challenges ...

An example problem

Consider the following example problem

$$-\nabla \cdot (a(x) \nabla u(x)) = f(x) \quad \text{in } D,$$
$$u(x) = 0 \quad \text{on } \partial D,$$

where $a(x)$ is the permeability, $f(x)$ is the source and $u(x)$ is the solution.

What if the input data is random?

We get a stochastic problem

$$-\nabla \cdot (a(\omega, x) \nabla u(\omega, x)) = f(\omega, x) \quad \text{in } \Omega \times D,$$
$$u(\omega, x) = 0 \quad \text{on } \Omega \times \partial D,$$

where $a(\omega, x)$, $f(\omega, x)$ and $u(\omega, x)$ are now random fields.

Challenge: Instead of just the mean-value solution $u(x_0)$ we would like also to know $\mathbb{E}[u](x_0)$, $\mathbb{V}ar[u](x_0)$ or even $\mathbb{P}[u(\omega, x_0) \geq u_0]$.

Random fields

A random field (RF) $a(\omega, x)$ is defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and indexed by a deterministic domain D:

- a set of RVs indexed by $x \in D$: for every $x \in D$, $a(\cdot, x)$ is a RV on Ω.
- a function-valued RV: for every $\omega \in \Omega$, $a(\omega, \cdot)$ is a realization of the RF in the domain D.

Mean

$$\bar{a}(x) = \mathbb{E}[a](x) = \int_\Omega a(\omega, \cdot)\, d\mathbb{P}(\omega),$$

and variance

$$\mathbb{V}ar[a](x) = \mathbb{E}\left[(\tilde{a})^2\right](x)$$

as a function of x with fluctuation (noise) part $\tilde{a}(\omega, x) = a - \bar{a}$.

Often only **second order information** (mean and covariance) are known.

Assumption: finite dimensional "noise"

Assume random fields (RFs) $a(\omega, x)$, $f(\omega, x)$ depends on finite number of random variables (RVs) $\mathbf{Y}(\omega) = [Y_1(\omega), \ldots, Y_N(\omega)] : \Omega \to \mathbb{R}^N$:

$$a_N(\omega, x) = a(\mathbf{Y}(\omega), x), \quad f_N(\omega, x) = f(\mathbf{Y}(\omega), x)$$

Motivation: piecewise constant material properties

Let $\{D_n\}_{n=1}^N$ be a partition of the spatial domain D
then define $a_N(\omega, x) = \sum_{i=1}^N \sigma_i Y_i(\omega) \chi_{D_i}(x)$.

Procedure: $\infty-$dimensional random field suitably truncated

- the interaction between points is described by a covariance function, e.g., $\mathbb{C}ov[a] = (x_1, x_2) = \mathbb{E}[\tilde{a}(\cdot, x_1)\tilde{a}(\cdot, x_2)] = \sigma^2 \exp\left(-\frac{\|x_1 - x_2\|^2}{L_c^2}\right)$

Expand a in a Karhunen-Loève expansion and retain the first N terms, denoted a_N, to capture most of the variability.

Properties of the covariance function

Let $a(\omega, x)$ be a RF with continuous covariance $\mathbb{C}_a : D \times D \to \mathbb{R}$

- $\mathbb{C}_a$ is symmetric, i.e., $\mathbb{C}_a(x_1, x_2) = \mathbb{C}_a(x_2, x_1), \quad \forall x_1, x_2 \in D$
- $\mathbb{C}_a$ is non-negative definite, i.e., $v^T \mathbb{C}_a(x, x) v \geq 0, \quad \forall v, x.$

Define the associated linear covariance operator $T_{\mathbb{C}_a} : L^2(D) \to L^2(D)$ by

$$[T_{\mathbb{C}_a} f](x_1) = \int_D \mathbb{C}_a(x_1, x_2) f(x_2) \, dx_2, \quad \forall f \in L^2(D).$$

Then $T_{\mathbb{C}_a} f \in C^0(D)$, $\forall f \in L^2(D)$, $\mathbb{C}_a \mapsto \mathbb{T}_{\mathbb{C}_a}$ is injective and $T_{\mathbb{C}_a}$ is compact, symmetric and non-negative definite:

- it has a countable sequence of real eigenvalues $\{\lambda_n\} \subset \mathbb{R}_+$, $\lambda \to 0$
- corresponding eigenfunctions $\{b_n(x)\}$ are $L^2(D)-$orthonormal

Karhunen-Loève expansion

a Fourier-type series based on the spectral expansion of covariance func.

$$a(\omega, x) = \bar{a}(x) + \sum_{n=1}^{\infty} \sqrt{\lambda_n} b_n(x) Y_n(\omega),$$

and $(\lambda_n, b_n(x))$ are eigenpairs of $T_{\mathbb{C}_a}$; $Y_n(\omega)$ are centered and uncorrelated RVs:

$$\mathbb{E}[Y_n] = 0, \quad \mathbb{C}ov[Y_n, Y_m] = \mathbb{E}[Y_n Y_m] = \delta_{nm},$$

but not necessarily independent.
We truncate the series

$$a(\omega, x) \approx a_N(\omega, x) = \bar{a} + \sum_{n=1}^{N} \sqrt{\lambda_n} b_n(x) Y_n(\omega).$$

Rate of decay depends on the smoothness of $\mathbb{C}_a$ and the corr. length L_c

Parametrization of random fields

Consider a random vector $\mathbf{Y}(\omega) = [Y_1(\omega), \ldots, Y_N(\omega)] : \Omega \to \mathbb{R}^N$ and define:

- $\Gamma_n \equiv Y_n(\Omega) \subset \mathbb{R}$ and $\Gamma = \Pi_{n=1}^N \Gamma_n \subset \mathbb{R}^N$ - image of the random vector $\mathbf{Y}(\Omega)$
- $\rho : \Gamma \to \mathbb{R}_+$ with $\rho \in L^\infty(\Gamma)$ as a joint PDF of $\mathbf{Y}(\omega)$, i.e., if $\mathbf{y} \in \Gamma$, and $\rho(\mathbf{y}) = \Pi_{n=1}^N \rho_n(y_n)$ for all $n, y_n \in \Gamma_n$, then

$$\mathbb{P}[Z \in \gamma \subset \Gamma] = \int_\gamma \rho(\mathbf{y}) \, d\mathbf{y},$$

which is a transformation of the measure $\mathbb{P}$ defined on Ω to $\mathbb{R}^N$.

Remark: curse of dimensionality when N is large

Approximating a stochastic PDE

- Given a description of $a_N(\mathbf{Y}(\omega), x)$ and ev. of $f_N(\mathbf{Y}(\omega), x)$, we would like to find $u_N(Y_1(\omega), \ldots, Y_N(\omega), x)$ such that

$$\mathcal{L}(a_N) u_N = f_N \quad \text{in } D \text{ a.s.}$$

Quantities of interest (QoI)

Our goal of predicting the statistical behavior of a physical system often requires the approximation of multi-dimensional statistical QoI, e.g:

$$\mathbb{E}[u](x) = \int_\Gamma u(\mathbf{y}, x) \rho(\mathbf{y}) \, d\mathbf{y}, \quad \text{where } \mathbf{y} \in \Gamma^N \quad \text{and } x \in \overline{D}$$

An application to a linear elliptic SPDE

Strong formulation:

find $u(\mathbf{y}, x)$ such that

$$\begin{aligned} -\nabla \cdot (a(\mathbf{y}, x) \nabla u(\mathbf{y}, x)) &= f(\mathbf{y}, x) \quad \text{for a.e. } x \in D, \\ u(\mathbf{y}, x) &= 0 \quad \text{for a.e. } x \in \partial D, \end{aligned}$$

where $\mathbf{y} \in \Gamma \subset \mathbb{R}^N$ and $x \in \overline{D}$

Next, define $V_\rho = L^2_\rho(\Gamma) \otimes H^1_0(D)$

Weak formulation:

find $u \in V_\rho$ such that $\forall v \in V_\rho$,

$$\mathbb{E}\left[\int_D a(\mathbf{y}, x) \nabla u(\mathbf{y}, x) \cdot \nabla v(\mathbf{y}, x) \, dx\right] = \mathbb{E}\left[\int_D f(\mathbf{y}, x) v(\mathbf{y}, x) \, dx\right]$$

Stochastic FEM: Stochastic Sampling Methods

Remark: The spatial discretization using finite element methods (FEM).

Stochastic Sampling Methods (SSMs):
randoms samples in Γ of PDE inputs are used to compute ensemble averages of statistical QoIs, e.g., Monte-Carlo FEM - *non-intrusive*

pros:

- allow reusability of deterministic codes
- the convergence rate is independent of the regularity of the solution u
- (and dimension with MC methods)

cons:

- do not yield fully discrete approximations
- slow conv. rates do not exploit the possible regularity of the solution

Stochastic FEM: Stochastic Polynomial Approximation

Observation: The analyticity of the solution $u(\mathbf{y}, x)$ wrt each random direction y_n suggests the use of (multivariate) polynomial approximation.

Idea

Approximate the response $u(\mathbf{y}, \cdot)$ by multi-variate global polynomials. The numer. solution should converge quickly since the solution is analytic in $\mathbf{y}$.

Stochastic polynomial approximation:

- Stochastic Collocation Methods (SCMs):
  probabilistic discretization is effected by collocating the FE solution on a particular set of points and then connecting the realizations with a suitable interpolating basis (Lagrangean) - *non-intrusive*
- Stochastic Galerkin Methods (SGMs):
  probabilistic discretization is effected by a spectral Galerkin projection onto, e.g., an L^2_ρ−orthogonal basis (Wiener or polynomial chaos) - *intrusive*

Approximating spaces

- Let $\mathcal{T}_h$ be a triangulation of D and $W_h(D) \subset W(D)$ contains cont. piecewise polynomials defined in $\mathcal{T}_h$ Assume $J = \dim[W_h(D)]$ and $\{\phi_j(x)\}_{j=1}^J \subset W_h$ is a FE basis for the deterministic domain
- Let $p = (p_1, \ldots, p_N)$ be a multi-index, $\mathcal{J}(p) \subset \mathbb{N}^N$ a multi-index set, with $p \in \mathbb{N}_+$ and define:

Multivariate polynomial space

$$\mathcal{P}_{\mathcal{J}(p)}(\Gamma) = \text{span}\left\{\Pi_{n=1}^N y_n^{p_n}, \quad \text{with } p \in \mathcal{J}(p)\right\} \subset L^2_\rho(\Gamma)$$

Assume $M = \dim\left[\mathcal{P}_{\mathcal{J}(p)}(\Gamma)\right]$ and $\{\psi_k\}_{k=1}^M$ form a basis for $\mathcal{P}_{\mathcal{J}(p)}(\Gamma)$, e.g. multivariate Legendre, Hermite, Lagrange, etc.

(Hermite) Orthogonal polynomials

The L^2_ρ−orthogonal basis was originally developed to approximate white noise processes with Gaussian measure [Wiener, 1938].

Let $\left\{H^{(n)}_{p_n}\right\}_{p_n=0}^P$ be the set of univariate Hermite polynomials (deg. $\leq P$) defined in $L^2_{\rho_n}(\Gamma_n)$ that are orthonormal wrt the Gaussian PDF $\rho_n(y_n)$ for each $n = 1, \ldots, N$:

$$\int_{\Gamma_n} H^{(n)}_{p_n}(y_n) H^{(n)}_{r_n}(y_n) \rho_n(y_n)\, dy_n = \delta_{p_n r_n}, \quad p_n, r_n \in \{0, \ldots, P\}$$

The multivariate $L^2_\rho(\Gamma)$ −orthogonal Hermite basis is defined as a tensor-product of univariate polynomials with $p \in \mathcal{J}(P)$:

$$H_p(y) = \Pi_{n=1}^N H^{(n)}_{p_n}(y_n), \quad \text{s.t. } \rho(y) = \Pi_{n=1}^N \rho_n(y_n),$$

where $\rho(y)$ is the Gaussian joint-PDF.

The Askey scheme: different PDFs and orthogonal polynomials

Identical construction for other orthogonal bases → generalized PC (gPC).
Example: uniform RVs → Legendre polynomial basis, etc.

distribution	polynomial type	support
Normal	Hermite	$(-\infty, \infty)$
Uniform	Legendre	$[-1, 1]$
Beta	Jacobi	$[-1, 1]$
Gamma	Generalized Laguerre	$[0, \infty)$
Exponential	Laguerre	$[0, \infty)$

Fully discrete approximation

We would like to find $u_p \in \mathcal{P}_{\mathcal{J}(p)}(\Gamma) \otimes W_h(D)$ such that

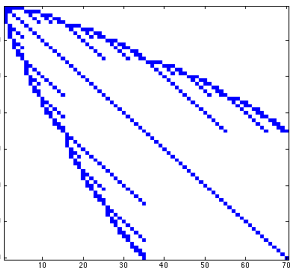
$$u_p(\mathbf{y}, x) = \sum_{j=1}^J \sum_{k=1}^M c_{jk} \phi_j(x) \psi_k(\mathbf{y}) = \sum_{k=1}^M u_k(x) \psi_k(\mathbf{y}), \quad u_k(x) \in W_h(D)$$

To compute the fully discrete approximation using SFEMs requires the resolution of the coefficients u_k which can be accomplished via:

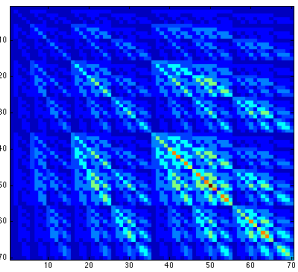
- non-intrusive methods by de-coupling the above expression and solving a M systems of size J ⟶ Stochastic Collocation Methods
  pros: de-coupling
  cons: possibility of integration and interpolation errors (aliasing).
- intrusive methods by solving the fully coupled $JM \times JM$ system ⟶ Stochastic Galerkin Methods
  pros: optimality of Galerkin projections
  cons: Implementation - requires development of new solvers.

Iterative solvers: motivation

Iterative solution and **preconditioning** of systems of linear equations, with a typical block structure given as:



(a) block sparse matrix



(b) block dense matrix

Outline

- Model problem and its discretization

Block sparse matrices:

- Structure of the global stochastic Galerkin matrix
- Hierarchical Schur complement preconditioner
- Numerical experiments (uniform random field)

Block dense matrices:

- Variant of the preconditioner for block dense matrices
- Numerical experiments (lognormal random field)

Model problem

Let $D \subset \mathbb{R}^d$, $d = 2, 3$, and let $(\Omega, \mathcal{F}, \mu)$ be a complete probability space.

We would like to find a function $u(x,\omega) : \overline{D} \times \Omega \to \mathbb{R}$ satisfying a.s.

$$\begin{aligned} -\nabla \cdot (k(x,\omega)\, \nabla u(x,\omega)) &= f(x) \qquad \text{in } D \times \Omega \\ u(x,\omega) &= 0 \qquad \text{on } \partial D \times \Omega \end{aligned}$$

Note: ∇ denotes the differentiation with respect to the spatial variables

Here $k(x,\omega)$ is a random scalar field such that

$$\mu\left(\omega \in \Omega : 0 < k_{\min} \le k(x,\omega) \le k_{\max} \quad \forall x \in \overline{D}\right) = 1$$

Next, let us introduce

$$U = H_0^1(D) \otimes L^2(\Omega), \qquad \|u\|_U = \sqrt{\mathbb{E}\left[\int_D |\nabla u|^2 \, dx\right]}.$$

Model problem: variational formulation

In the weak formulation we would like to solve

$$u \in U: \; a(u,v) = \langle f, v \rangle, \qquad \forall v \in U$$

where

$$a(u,v) = \mathbb{E}\left[\int_D k(x,\omega)\, \nabla u \cdot \nabla v \, dx\right], \qquad \langle f, v \rangle = \mathbb{E}\left[\int_D f\, v \, dx\right]$$

We assume that k has a Karhunen-Loève (KL) expansion

$$k(x,\omega) = \sum_{i=0}^{N} k_i(x)\, \xi_i(\omega) \quad \xi_0 = 1, \quad \xi_i \sim \mathrm{U}[0,1] \quad i > 0$$

further assuming $\xi_i(\omega)$ to be i.i.d. random variables. We consider

$$u = \sum_{j=0}^{M} u_j \psi_j(\xi_0, \ldots, \xi_N).$$

Model problem: stochastic finite element discretization

Finite element discretization yields

$$\sum_{j=0}^{M} \sum_{i=0}^{N} c_{ijk} K_i u_j = f_k, \qquad k = 0, \ldots, M,$$

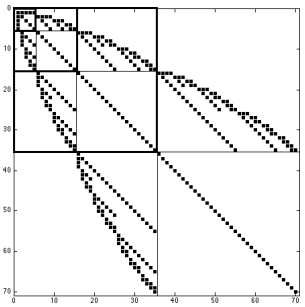
where $c_{ijk} = \mathbb{E}[\xi_i \psi_j \psi_k]$ or more generally $c_{ijk} = \mathbb{E}[\psi_i \psi_j \psi_k]$. Defining

$$K^{(j,k)} = \sum_{i=0}^{N} c_{ijk} K_i$$

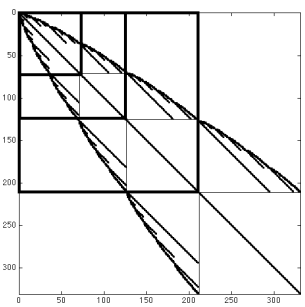
the global system can be written as

$$\begin{bmatrix} K^{(0,0)} & K^{(0,1)} & \cdots & K^{(0,M)} \\ & \ddots & & \\ \vdots & & K^{(k,k)} & \vdots \\ & & & \ddots \\ K^{(M,0)} & K^{(M,1)} & \cdots & K^{(M,M)} \end{bmatrix} \begin{bmatrix} u_0 \\ \vdots \\ u_k \\ \vdots \\ u_M \end{bmatrix} = \begin{bmatrix} f_0 \\ \vdots \\ f_k \\ \vdots \\ f_M \end{bmatrix}.$$

Matrix hierarchy



$N = 4$, $P = 4 \to 350$ blocks



$N = 4$, $P = 7 \to 2010$ blocks

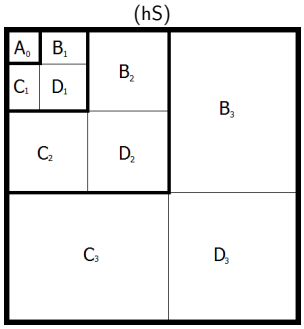
N ... stochastic dimension

P ... order of the polynomial expansion

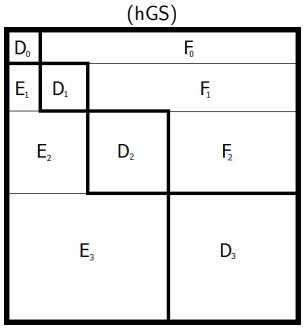
Matrix hierarchy: two points of view

hierarchical ...

... Schur complement preconditioner (hS)



... Gauss-Seidel preconditioner (hGS)



Matrix hierarchy (towards the Schur complement preconditioner)

Let us write the global problem as

$$A_P u_P = f_P$$

where the matrix has a recursive structure

$$A_\ell = \begin{bmatrix} A_{\ell-1} & B_\ell \\ C_\ell & D_\ell \end{bmatrix}, \qquad \ell = P, \dots, 1$$

- ℓ ... corresponding to the $\ell-$th degree stoch. polynom. expansion
- block D_ℓ is block diagonal for all ℓ

The mean-value problem is

$$A_0 u_0 = f_0$$

Idea of the preconditioner

By block LU decomposition

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} I_A & BD^{-1} \\ 0 & I_D \end{bmatrix} \begin{bmatrix} S & 0 \\ 0 & D \end{bmatrix} \begin{bmatrix} I_A & 0 \\ D^{-1}C & I_D \end{bmatrix}$$

where $S = A - BD^{-1}C$ is the Schur complement with respect to D.

Inverting

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} I_A & 0 \\ -D^{-1}C & I_D \end{bmatrix} \begin{bmatrix} S^{-1} & 0 \\ 0 & D^{-1} \end{bmatrix} \begin{bmatrix} I_A & -BD^{-1} \\ 0 & I_D \end{bmatrix}$$

Idea of the preconditioner:
1. replace S^{-1} by A^{-1}
2. use this block inverse throughout the hierarchy of the global matrix

Idea of the preconditioner: matrix hierarchy

In the action of the preconditioner we would like to approximate

$$A_P u_P = f_P$$

written as

$$\begin{bmatrix} A_{P-1} & B_P \\ C_P & D_P \end{bmatrix} \begin{bmatrix} u_P^{P-1} \\ u_P^P \end{bmatrix} = \begin{bmatrix} f_P^{P-1} \\ f_P^P \end{bmatrix}$$

by

$$A_P^{-1} \approx \begin{bmatrix} I_A & 0 \\ -D_P^{-1}C_P & I_D \end{bmatrix} \begin{bmatrix} \mathbf{A_P^{-1}_{-1}} & 0 \\ 0 & D_P^{-1} \end{bmatrix} \begin{bmatrix} I_A & -B_P D_P^{-1} \\ 0 & I_D \end{bmatrix}$$

and replace inverse of $S = A - BD^{-1}C$ by inverse only of A as

$$S_{\ell-1}^{-1} \approx A_{\ell-1}^{-1} \qquad \ell = P-1, \dots, 1.$$

Hierarchical Schur complement preconditioner

The preconditioner $M_P : r_P \longmapsto u_P$ is defined as follows:
for $\ell = P, \dots 1$,

split the residual as $r_\ell = \left[r_\ell^{\ell-1}, r_\ell^\ell\right]$

compute the **pre-correction** as

$$g_{\ell-1} = r_\ell^{\ell-1} - B_\ell D_\ell^{-1} r_\ell^\ell$$

if $\ell > 1$, set $r_{\ell-1} = g_{\ell-1}$, else solve $A_0 u_0 = g_0$.

end
for $\ell = 1, \dots P$,

compute the **post-correction**, i.e., set $u_\ell^{\ell-1} = u_{\ell-1}$, solve

$$u_\ell^\ell = D_\ell^{-1}\left(r_\ell^\ell - C_\ell u_\ell^{\ell-1}\right)$$

and concatenate $u_\ell = \left[u_\ell^{\ell-1}, u_\ell^\ell\right]$.

If $\ell < P$, set $u_{\ell+1}^\ell = u_\ell$.

end

Implementation

The main computational work is in the application of D_ℓ^{-1}, $\ell = P, \dots, 1$.

Note:
- all D_ℓ are block diagonal matrices
- each block of D_ℓ has the size of the underlying deterministic problem
- the blocks are closely related to $A_0 = K_0$ (the mean-value problem).

Idea:
Replace the exact solves of D_ℓ with iterative block solves
(independent inner Krylov iterations, using a preconditioner $M_0 \approx A_0$).

Block count: in the action of the Schur complement preconditioner

N ... stochastic dimension, P ... order of polynomial expansion
(one is changing, the other one is set to four)

N or P	n_b	n_{db}	n_m	n_{ds}
1	13	5	8	9
2	55	15	40	29
3	155	35	120	69
4	350	70	280	139
5	686	126	560	251
6	1218	210	1008	419
7	2010	330	1680	659
8	3135	495	2640	989

n_b ... total number of blocks
n_{db} ... number of diagonal blocks
n_m ... number of block matrix-vector multiplications
n_{ds} ... number of its block diagonal solves

Numerical results: increasing the stochastic dimension

Poisson's eq. in $[0,1]^2$, 10×10 finite elements, uniform r.f., $CoV = 50\%$,
$N \ldots$ stochastic dimension, $P \ldots$ order of polynom. expansion ($P = 4$),
mb ...mean-based, bGS ...block Gauss-Seidel, hS ...hierarchical Schur prec.,

setup		mb		bGS		hS	
N	$ndof$	$iter$	κ	$iter$	κ	$iter$	κ
1	605	12	2.0127	5	1.0507	5	1.0465
2	1815	15	2.7340	6	1.1279	6	1.1236
3	4235	16	2.9995	7	1.1693	6	1.1514
4	8470	17	3.3413	7	1.2131	7	1.2028
5	15,246	18	3.5891	7	1.2447	7	1.2434
6	25,410	18	3.6349	7	1.2501	7	1.2559
7	39,930	19	4.0993	8	1.3202	7	1.3146
8	59,895	19	4.0597	8	1.3198	7	1.3182

$ndof \ldots$ degrees of freedom of the global stochastic Galerkin matrix,
$iter \ldots$ CG iterations (tol 10^{-8}), $\kappa \ldots$ cond. number estimate.

Numerical results: increasing the polynomial degree

Poisson's eq. in $[0,1]^2$, 10×10 finite elements, uniform r.f., $CoV = 50\%$,
$N \ldots$ stochastic dimension ($N = 4$), $P \ldots$ order of polynom. expansion,
mb ...mean-based, bGS ...block Gauss-Seidel, hS ...hierarchical Schur prec.,

setup		mb		bGS		hS	
P	$ndof$	$iter$	κ	$iter$	κ	$iter$	κ
1	605	9	1.6391	5	1.0626	5	1.0624
2	1815	13	2.2379	6	1.1117	6	1.1109
3	4235	15	2.8122	7	1.1658	6	1.1559
4	8470	17	3.3413	7	1.2131	7	1.2028
5	15,246	18	3.7824	7	1.2538	7	1.2426
6	25,410	19	4.1534	8	1.2921	7	1.2798
7	39,930	20	4.4708	8	1.3219	7	1.3125
8	59,895	20	4.7371	8	1.3472	7	1.3398

$ndof \ldots$ degrees of freedom of the global stochastic Galerkin matrix,
$iter \ldots$ CG iterations (tol 10^{-8}), $\kappa \ldots$ cond. number estimate.

Numerical results: increasing the Coefficient of Variation

Poisson's equation in $[0,1]^2$, 10×10 elements, uniform random field,
$N \ldots$ stoch. dim., $P \ldots$ order of polynom. expansion ($N = P = 4$),
mb ...mean-based, bGS ...block Gauss-Seidel, hS ...hierarchical Schur prec.,

setup	mb		bGS		hS	
CoV (%)	$iter$	κ	$iter$	κ	$iter$	κ
5	6	1.0960	3	1.0008	3	1.0009
15	9	1.3514	4	1.0090	4	1.0089
25	11	1.7021	5	1.0314	5	1.0304
35	13	2.1808	6	1.0770	5	1.0664
45	16	2.8773	6	1.1510	6	1.1414
55	19	3.9523	8	1.2948	7	1.2830

$CoV = \mu/\sigma \ldots$ the Coefficient of Variation (%),
$ndof \ldots$ size of the global stochastic Galerkin matrix is 8470.

Numerical results: decreasing the mesh size

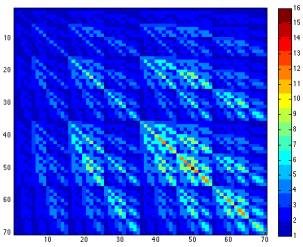
Poisson's equation in $[0,1]^2$, 10×10 elements, uniform random field,
$N \ldots$ stoch. dim., $P \ldots$ order of polynom. expansion ($N = P = 4$),
mb ...mean-based, bGS ...block Gauss-Seidel, hS ...hierarchical Schur prec.,

setup		mb		bGS		hS	
h	$ndof$	$iter$	κ	$iter$	κ	$iter$	κ
1/5	2520	16	3.2484	7	1.2022	6	1.1790
1/10	8470	17	3.3413	7	1.2131	7	1.2028
1/15	17920	17	3.3145	7	1.2063	7	1.2047
1/20	30870	17	3.3463	7	1.2110	7	1.2032
1/25	47320	17	3.3473	7	1.2112	7	1.2032
1/30	67270	17	3.3190	7	1.2070	7	1.2054

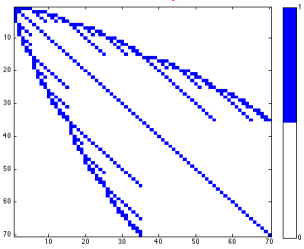
$ndof \ldots$ size of the global stochastic Galerkin matrix.

Structure of the matrix (block dense case)

structure of the matrix



Q: structure of the preconditioner?

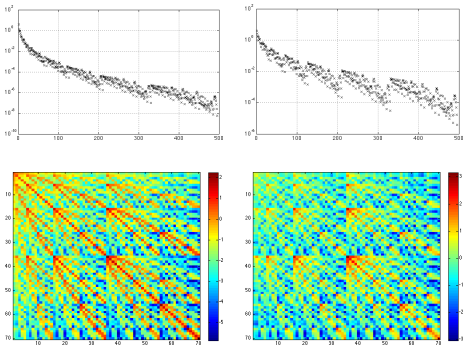


$$K^{(j,k)} = \sum_{i=0}^{M_k} c_{ijk} K_i \qquad \text{MAT-VEC: } v_{(j)} = \sum_{k=0}^{M} \sum_{i=0}^{M_k} c_{ijk} K_i u_{(k)}$$

Theory (Matthies & Keese, 2005): $M_k >> M$.

Stiffness matrices: decay of norm(K_i)

Plots: norm(K_i) and $c^{(j,k)} = \sum_i c_{ijk} \cdot \text{norm}(K_i)$, where $c_{ijk} = \mathbb{E}[\psi_i \psi_j \psi_k]$.



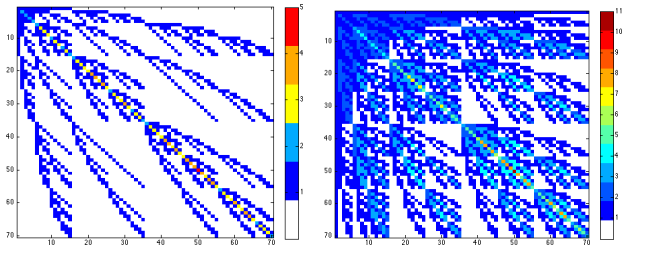
$CoV = 50\%$ $CoV = 150\%$

Modification 1: truncated preconditioners

Idea: in the action of the preconditioner, replace the MAT-VEC operation

$$v_{(j)} = \sum_{k=0}^{M} \sum_{i=0}^{M_k} c_{ijk} K_i u_{(k)} \quad \text{by} \quad v_{(j)} = \sum_{k=0}^{M} \sum_{i \in \mathcal{M}_t} c_{ijk} K_i u_{(k)},$$

$\mathcal{M}_t \ldots$ adaptively selected subset of indexes from the set $\{0, 1, \ldots, M_k\}$.

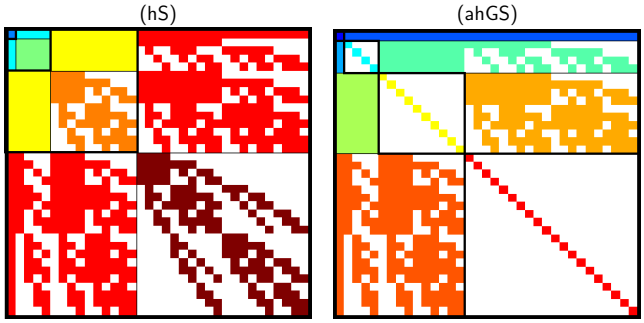


Modification 2: approximate preconditioners

Idea: approximate the solves with submatrices D_ℓ by diagonal block solves.

Schur complement preconditioner (hS)

approximate Gauss-Seidel prec. (ahGS)



Numerical results: increasing the stochastic dimension

full vs. approximate preconditioners (no truncation yet)

Poisson's eq., $[0,1]^2$, 10×10 finite elements, lognormal r.f., $CoV = 100\%$,
$N \ldots$ stoch. dim., $P \ldots$ order of polynom. expansion ($P = 4, P_k = 2P$),

setup	mb		hS		ahS		bGS		ahGS	
N	*iter*	κ	*iter*	κ	*iter*	κ	*iter*	κ	*iter*	κ
1	48	28.76	15	3.40	15	3.40	15	3.42	15	3.42
2	61	37.16	16	3.62	27	8.06	17	3.75	16	3.45
3	62	38.07	16	3.76	31	10.77	17	3.74	18	4.35
4	66	43.65	16	4.17	38	15.28	19	4.29	19	4.74

ndof ... degrees of freedom of the global stochastic Galerkin matrix,
iter ... CG iterations (tol 10^{-8}), $\kappa \ldots$ cond. number estimate.

Numerical results: increasing the polynomial degree

full vs. approximate preconditioners (no truncation yet)

Poisson's eq., $[0,1]^2$, 10×10 finite elements, lognormal r.f., $CoV = 100\%$,
$N \ldots$ stoch. dim. ($N = 4$), $P \ldots$ order of polynom. expansion, $P_k = 2P$,

setup	mb		hS		ahS		bGS		ahGS	
P	*iter*	κ	*iter*	κ	*iter*	κ	*iter*	κ	*iter*	κ
1	15	3.50	7	1.39	11	1.76	8	1.39	8	1.31
2	28	8.95	10	1.93	16	3.04	12	1.97	11	1.76
3	44	20.04	13	2.80	24	6.09	15	2.87	14	2.58
4	66	43.65	16	4.17	38	15.28	19	4.29	19	4.74

ndof ... degrees of freedom of the global stochastic Galerkin matrix,
iter ... CG iterations (tol 10^{-8}), $\kappa \ldots$ cond. number estimate.

Numerical results: increasing the Coefficient of Variation

full vs. approximate preconditioners (no truncation yet)

Poisson's eq., $[0,1]^2$, 10×10 finite elements, lognormal r.f., $CoV = 100\%$,
$N \ldots$ stoch. dim., $P \ldots$ order of polynom. expan. ($N = P = 4, P_k = 2P$),

	mb		hS		ahS		bGS		ahGS	
CoV	*iter*	κ	*iter*	κ	*iter*	κ	*iter*	κ	*iter*	κ
25	16	3.24	7	1.18	8	1.25	7	1.18	6	1.12
50	29	9.36	10	1.78	14	2.45	11	1.77	10	1.62
75	46	22.21	13	2.85	23	6.01	15	2.82	14	2.69
100	66	43.65	16	4.17	38	15.28	19	4.29	19	4.74
125	85	72.76	19	5.54	58	36.34	23	5.98	26	8.21
150	103	107.07	21	6.85	84	77.73	26	7.75	35	13.74

CoV (%) ... the Coefficient of Variation,
ndof ... size of the global stochastic Galerkin matrix is 8470.

Numerical results: decreasing the mesh size

Poisson's equation in $[0,1]^2$, 10×10 elements, uniform random field,
$N \ldots$ stoch. dim., $P \ldots$ order of polynom. expansion ($N = P = 4$),
mb ...mean-based, hS ...hierarchical Schur prec.,
bGS ...block Gauss-Seidel, ahGS ...approximate hierarchical Gauss-Seidel,

setup		mb		hS		bGS		ahGS	
h	*ndof*	*iter*	κ	*iter*	κ	*iter*	κ	*iter*	κ
1/5	2520	59	40.62	15	3.84	18	3.99	19	4.91
1/10	8470	66	43.65	16	4.17	19	4.29	19	4.74
1/15	17920	68	44.42	16	4.24	19	4.38	20	4.72
1/20	30870	69	44.89	17	4.25	19	4.37	20	4.78
1/25	47320	69	44.94	17	4.26	20	4.40	20	4.81
1/30	67270	71	45.11	17	4.26	19	4.37	20	4.75

ndof ... size of the global stochastic Galerkin matrix.

Numerical results: truncation of the MAT-VEC

N ... stoch. dim., P ... order of polynom. expansion ($N = P = 4, P_k = 2P$).
Drop matrices corresponding to higher order expansion of the coefficient than ℓ_t.

setup			hS		ahS		bGS		ahGS	
ℓ_t	M_t+1	$nz(c_{ijk})$	*iter*	κ	*iter*	κ	*iter*	κ	*iter*	κ
		$CoV = 25\%$	(mb:	*iter* $= 16$		$\kappa = 3.24$)				
0	1	70	16	3.20	16	3.19	16	3.19	16	3.19
1	5	350	8	1.27	8	1.33	7	1.23	7	1.23
2	15	1210	7	1.21	8	1.25	7	1.20	6	1.17
4	70	4980	7	1.18	8	1.25	7	1.18	6	1.12
8	495	12585	7	1.18	8	1.25	7	1.18	6	1.12
		$CoV = 150\%$	(mb:	*iter* $= 103$		$\kappa = 107.07$)				
0	1	70	71	61.44	89	90.18	89	90.18	89	90.18
1	5	350	51	29.92	59	39.66	57	36.85	57	36.85
2	15	1210	51	30.18	60	42.06	46	24.26	50	27.53
4	70	4980	32	12.05	58	38.08	28	9.42	34	13.86
8	495	12585	21	6.85	84	77.73	26	7.75	35	13.74

Numerical results: adaptive truncation of the MAT-VEC

N ... stoch. dim., P ... order of polynom. expansion ($N = P = 4, P_k = 2P$).
Adaptively drop matrices for which $\max_{jk}(c_{ijk}) \cdot \mathrm{norm}(K_i) < \tau$.

setup			hS		bGS		ahGS	
τ	N_{adapt}	$nz(c_{ijk})$	*iter*	κ	*iter*	κ	*iter*	κ
		$CoV = 25\%$	mb:	*iter* $= 16$	$\kappa = 3.23556$			
10^{+1}	5	345	10	1.63	10	1.57	10	1.57
10^{0}	13	877	7	1.20	7	1.18	6	1.12
10^{-1}	32	2057	7	1.18	7	1.17	6	1.12
0	495	12585	7	1.18	7	1.18	6	1.12
		$CoV = 150\%$	mb:	*iter* $= 103$	$\kappa = 107.067$			
10^{+2}	10	336	66	50.2	70	50.68	70	50.68
10^{+1}	55	2450	30	10.89	29	9.49	25	6.95
10^{0}	171	6338	23	7.00	27	7.98	32	11.48
10^{-1}	313	9714	22	6.86	26	7.74	35	13.54
0	495	12585	21	6.85	26	7.75	35	13.74

Conclusion

A methodology of hierarchical preconditioning (Schur and Gauss-Seidel):

- approximation using the diagonal block solves
- truncation of the MAT-VEC operations

Advantages:

- neither the matrix, nor the preconditioner need to be formed explicitly
- the ingredients include only
  - the stiffness matrices from the polynomial chaos expansion
  - a good preconditioner M_0 for the mean-value deterministic problem
- allows an obvious parallel implementation
- can be written as a "wrapper" around existing deterministic code (for the corresponding mean-value problem); and thus

minimally intrusive

Adaptivita pro lineární a nelineární řešiče a výběr časového kroku a prostorové sítě v numerických diskretizacích

Martin Vohralík

INRIA Paris-Rocquencourt

Seminář numerické analýzy, Rožnov pod Radhoštěm, 21.–25. ledna 2013

Outline

Numerical approximation of a nonlinear, unsteady PDE

Exact and approximate solution

- let u be the weak solution of $A(u) = f$, A **nonlinear**, **unsteady** partial differential equation (PDE) on $\Omega \times (0, T)$
- let $u_{h\tau}$ be its approximate numerical solution, $\mathcal{A}_{h\tau}(u_{h\tau}) = F_{h\tau}$

Solution algorithm

- introduce a temporal mesh of $(0, T)$ given by t^n, $0 \le n \le N$
- introduce a spatial mesh $\mathcal{T}_h^n$ of Ω on each t^n
- on each t^n, solve a system of nonlinear algebraic equations $\mathcal{A}_h^n(u_h^n) = F_h^n$

Iterative solvers and space and time steps choice

Iterative linearization of $\mathcal{A}_h^n(u_h^n) = F_h^n$ on each t^n

- $\mathbb{A}_h^{n,k-1} u_h^{n,k} = F_h^{n,k-1}$: discrete iterative linearization (Newton, fixed-point)
- loop in k
- **when do we stop?**

Iterative algebraic solver on each t^n and for each k

- $\mathbb{A}_h^{n,k-1} u_h^{n,k} = F_h^{n,k-1}$ is a linear algebraic system
- we only solve it inexactly by some iterative algebraic solver: loop in i
- **when do we stop?**

Temporal mesh

- choice of the discrete times t^n?

Spatial mesh

- choice of the meshes $\mathcal{T}_h^n$?

Optimal solution strategy

Optimal solution strategy

- give a **guaranteed** and **robust** upper bound on the overall error $\|u - u_{h\tau}\|_{\Omega\times(0,T)}$, as tight as possible
- **distinguish** the algebraic, linearization, temporal, and spatial error **components**
- **stop** the **iterative solvers** whenever the corresponding errors do not affect the overall error significantly
- **refine/derefine adaptively** the time and space **meshes** and **equilibrate** the space and time **errors**

Benefits

- **optimal computable overall error bound**
- **important computational savings**
- **improvement of approximation precision**

Previous results

Steady problems

- Babuška and Rheinboldt (1978), introduction of a posteriori estimates
- Ladevèze and Leguillon (1983), equilibrated fluxes estimates (equality of Prager and Synge (1947))
- Verfürth (1996), residual-based estimates
- Ainsworth (2005), nonconforming methods

Unsteady problems

- Bieterman and Babuška (1982), introduction
- Verfürth (2003), efficiency, robustness with respect to the final time
- Makridakis and Nochetto (2003), elliptic reconstruction

Previous results

Nonlinear problems

- Han (1994), general framework
- Verfürth (1994), residual estimates

Stopping criteria

- engineering literature, since 1950's
- Becker, Johnson, and Rannacher (1995), multigrid st. crit.
- Arioli (2000's), general linear solver st. crit.
- Chaillou and Suri (2006, 2007), distinguishing discretization and linearization errors

Model diffusion problem

Model diffusion problem
Let $\Omega \subset \mathbb{R}^d$, $d \geq 1$. Find $u : \Omega \to \mathbb{R}$ such that

$$\begin{aligned} -\nabla\cdot(\underline{\mathbf{K}}\nabla u) &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned}$$

where

- $\underline{\mathbf{K}} : \Omega \to \mathbb{R}^{d\times d}$ is a diffusion tensor,
- $f : \Omega \to \mathbb{R}$ is a source term.

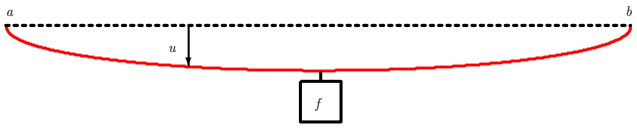
Form in 1D
Let $\Omega =]a, b[$, $a < b$. Let $k :]a, b[\to \mathbb{R}$ and $f :]a, b[\to \mathbb{R}$ be two given functions. Find $u :]a, b[\to \mathbb{R}$ such that

$$\begin{aligned} -(ku')' &= f, \\ u(a) = u(b) &= 0. \end{aligned}$$

Weak formulation
Find $u \in V := H_0^1(\Omega)$ such that

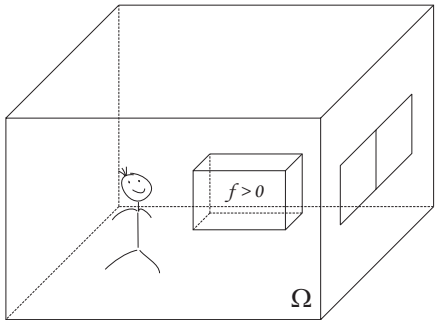
$$(\underline{\mathbf{K}}\nabla u, \nabla v) = (f, v) \qquad \forall v \in V.$$

Example: elastic string



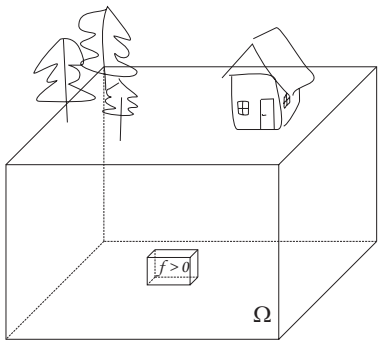
Elastic string with displacement u and weight f

Example: heat flow



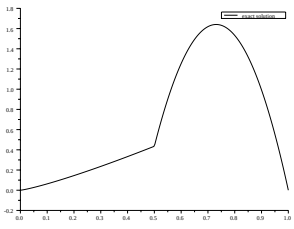
A room with a heater of $f > 0$ and temperature u

Example: underground water flow

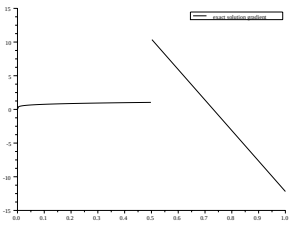


Underground with a water well of $f > 0$ and pressure head u

Properties of the exact solution

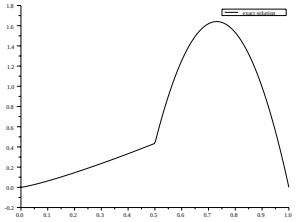


Solution u (displacement, temperature, pressure ...) is continuous

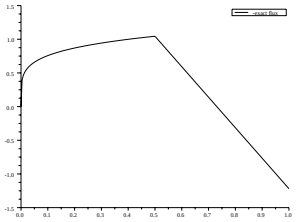


Solution gradient ∇u (derivative u' in 1D) is not necessarily continuous

Properties of the exact solution

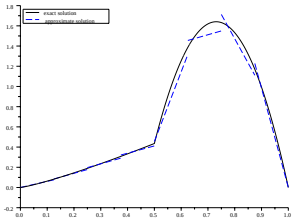


Solution u is continuous

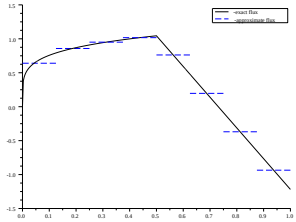


Flux $\mathbf{t} := -\underline{\mathbf{K}}\nabla u$ (or $-ku'$ in 1D) is continuous

Approximate solution and approximate flux

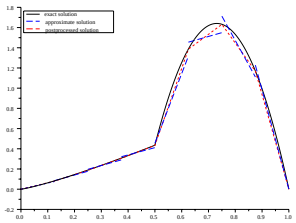


Approximate solution u_h is not necessarily continuous

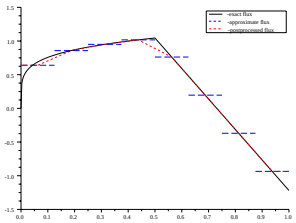


Approximate flux $-\underline{\mathbf{K}}\nabla u_h$ ($-ku_h'$) is not necessarily continuous

Potential and flux reconstructions



Potential reconstruction



Flux reconstruction

A posteriori error estimate for $-\nabla\cdot(\nabla u) = f$ ($\underline{\mathbf{K}} = \mathbb{I}$)

Assumption A (Equilibrated flux reconstruction)

There exists an **equilibrated flux reconstruction** $\mathbf{t}_h \in \mathbf{H}(\mathrm{div},\Omega)$ *s.t.*

$$(\nabla\cdot\mathbf{t}_h, 1)_K = (f,1)_K \qquad \forall K \in \mathcal{T}_h.$$

Assumption B (Potential reconstruction)

There exists a **potential reconstruction** $s_h \in V$.

Theorem (A guaranteed a posteriori error estimate)

Let

- $u \in V$ *be the weak solution,*
- $u_h \in V(\mathcal{T}_h) := \{v \in L^2(\Omega),\ v|_K \in H^1(K)\ \forall K \in \mathcal{T}_h\}$ *be arbitrary,*
- *Assumptions A and B hold.*

Then

$$\|\nabla(u-u_h)\|^2 \le \sum_{K\in\mathcal{T}_h} (\eta_{\mathrm{F},K} + \eta_{\mathrm{R},K})^2 + \sum_{K\in\mathcal{T}_h} \eta_{\mathrm{NC},K}^2,$$

where $\eta_{\mathrm{F},K}$, $\eta_{\mathrm{R},K}$, $\eta_{\mathrm{NC},K}$ *are fully computable from* u_h, $\mathbf{t}_h$, s_h.

Estimators

- *nonconformity estimator*

$$\eta_{\mathrm{NC},K} := \|\nabla(u_h - s_h)\|_K$$

  - evaluates the departure of u_h from V
  - constraint $u \in V$

- *flux estimator*

$$\eta_{\mathrm{F},K} := \|\nabla u_h + \mathbf{t}_h\|_K$$

  - evaluates the departure of ∇u_h from $\mathbf{H}(\mathrm{div},\Omega)$
  - constitutive law $\mathbf{t} = -\nabla u$ and constraint $\mathbf{t} \in \mathbf{H}(\mathrm{div},\Omega)$

- *residual estimator*

$$\eta_{\mathrm{R},K} := \frac{h_K}{\pi}\|f - \nabla\cdot\mathbf{t}_h\|_K$$

  - strong form of the PDE evaluated for the flux $\mathbf{t}_h$
  - equilibrium $\nabla\cdot\mathbf{t} = f$

Assumptions for efficiency

Assumption C (Technical assumption)

Let $\mathcal{T}_h$ *be shape-regular and* u_h, f, *and* $\mathbf{t}_h$ *pw polynomials.*

Assumption D (Potential reconstruction)

Let the potential reconstruction s_h *be a piecewise polynomial constructed from* u_h *by local averaging.*

Assumption E (Approximation property – flux reconstruction)

For all $K \in \mathcal{T}_h$, *there holds*

$$\eta_{\mathrm{F},K} \lesssim \eta_{\sharp,\mathfrak{T}_K},$$

where

$$\eta_{\sharp,\mathfrak{T}_K} := \left\{\sum_{K'\in\mathfrak{T}_K} h_{K'}^2 \|f + \Delta u_h\|_{K'}^2\right\}^{1/2} + \left\{\sum_{e\in\mathfrak{E}_K^{\mathrm{int}}} h_e \|[\![\nabla u_h]\!]\cdot\mathbf{n}_e\|_e^2\right\}^{1/2} + \left\{\sum_{e\in\mathfrak{E}_K} h_e^{-1}\|[\![u_h]\!]\|_e^2\right\}^{1/2}.$$

Local efficiency

Theorem (Local efficiency)

Let $\langle [\![u_h]\!], 1\rangle_e = 0$ for all faces e of the mesh $\mathcal{T}_h$. Then, under Assumptions C to E,

$$\eta_{\mathrm{NC},K} + \eta_{\mathrm{R},K} + \eta_{\mathrm{F},K} \lesssim \|\nabla(u - u_h)\|_{\mathfrak{T}_K}$$

for all $K \in \mathcal{T}_h$.

Summary for $-\nabla\cdot(\nabla u) = f$

Summary

- **guaranteed upper bound**:
$$\|\nabla(u - u_h)\| \leq \left\{ \sum_{K\in\mathcal{T}_h} \eta_K^2 \right\}^{1/2}$$
- **local efficiency**:
$$\eta_K \lesssim \|\nabla(u - u_h)\|_{\mathfrak{T}_K} \qquad \forall K \in \mathcal{T}_h$$
- close to **asymptotic exactness**:
$$\frac{\left\{ \sum_{K\in\mathcal{T}_h} \eta_K^2 \right\}^{1/2}}{\|\nabla(u - u_h)\|} \searrow 1$$
- **robustness**: the three previous properties hold independently of the parameters of the problem and of their variation (size of Ω, shape of Ω, regularity of u, local refinement of $\mathcal{T}_h$, sizes h_K)
- **small evaluation cost** of η_K

Application

Discretization methods

- conforming finite elements
- nonconforming finite elements
- discontinuous Galerkin method
- various finite volumes
- mixed finite elements

Application

- specification of the potential reconstruction s_h and flux reconstruction $\mathbf{t}_h$
- $s_h = u_h$ in conforming methods (FE, VCFV) $\Rightarrow \eta_{\mathrm{NC},K} = 0$
- $\mathbf{t}_h = -\underline{\mathbf{K}}\nabla u_h$ in flux-conforming methods (CCFV, MFE) $\Rightarrow \eta_{\mathrm{F},K} = 0$
- verification of Assumptions A to E

Numerics: finite elements in 1D

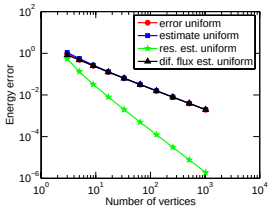
Model problem

$$\begin{aligned} -u'' &= \pi^2 sin(\pi x) \quad \text{in } (0,1), \\ u &= 0 \quad \text{in } 0,1 \end{aligned}$$

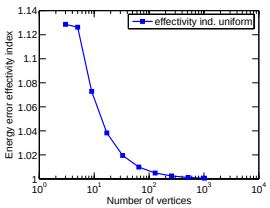
Exact solution

$$u(x) = sin(\pi x)$$

Estimated and actual errors, effectivity index



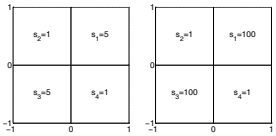
Actual error and estimator and its components



Effectivity index

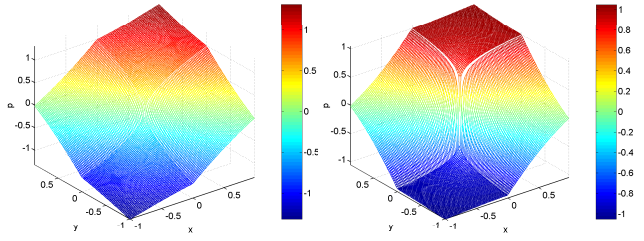
Numerics: cell-centered finite volumes

- diffusion equation
$$-\nabla\cdot(\underline{\mathbf{K}}\nabla u) = 0 \quad \text{in} \quad \Omega = (-1,1)\times(-1,1)$$
- discontinuous and inhomogeneous $\underline{\mathbf{K}}$, two cases:



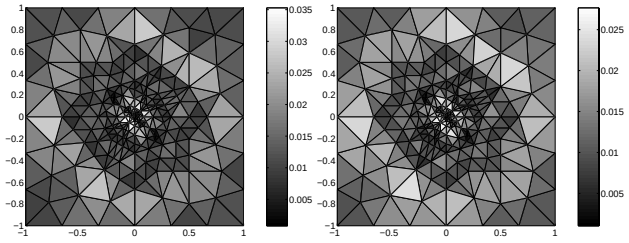
- analytical solution: singularity at the origin
$$u(r,\theta)|_{\Omega_i} = r^\alpha (a_i \sin(\alpha\theta) + b_i \cos(\alpha\theta))$$
  - (r,θ) polar coordinates in Ω
  - a_i, b_i constants depending on Ω_i
  - α regularity of the solution

Analytical solutions



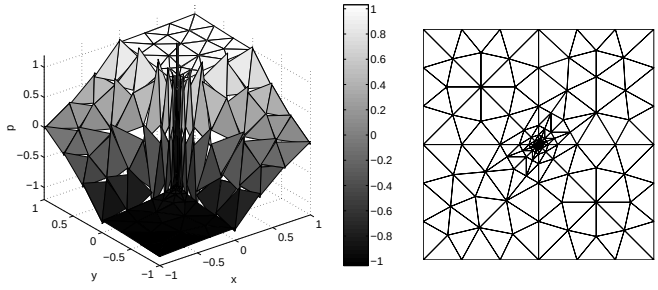
Case 1 Case 2

Error distribution on an adaptively refined mesh, case 1

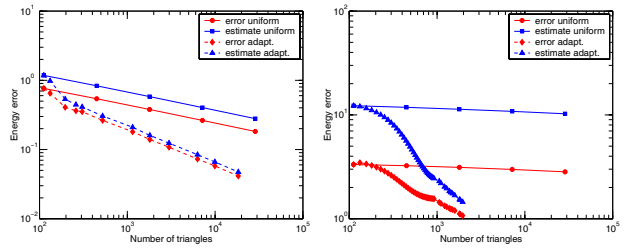


Estimated error distribution Exact error distribution

Approximate solution and the corresponding adaptively refined mesh, case 2

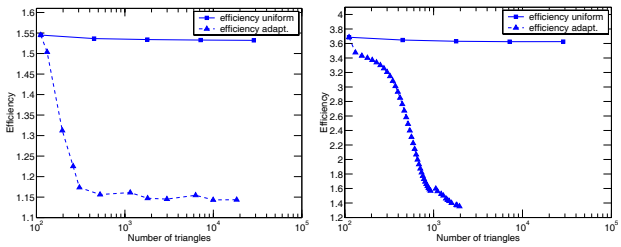


Estimated and actual errors in uniformly/adaptively refined meshes



Case 1 Case 2

Effectivity indices in uniformly/adaptively refined meshes



Case 1 Case 2

Inexact Newton method

System of nonlinear algebraic equations

Nonlinear operator $\mathcal{A}: \mathbb{R}^N \to \mathbb{R}^N$, vector $F \in \mathbb{R}^N$: find $U \in \mathbb{R}^N$ s.t.

$$\mathcal{A}(U) = F$$

Algorithm (Inexact linearization)

- *Choose initial vector* U^0. *Set* $k := 1$.
- $U^{k-1} \Rightarrow$ *matrix* $\mathbb{A}^{k-1}$ *and vector* F^{k-1}: *find* U^k *s.t.*

$$\mathbb{A}^{k-1} U^k \approx F^{k-1}.$$

  - *Set* $U^{k,0} := U^{k-1}$ *and* $i := 1$.
  - *Do* 1 *algebraic solver step* $\Rightarrow U^{k,i}$ *s.t.* ($R^{k,i}$ *algebraic res.*)

$$\mathbb{A}^{k-1} U^{k,i} = F^{k-1} - R^{k,i}.$$

  - *Convergence? OK* $\Rightarrow U^k := U^{k,i}$. *KO* $\Rightarrow i := i + 1$, *back to 3.2.*
- *Convergence? OK* $\Rightarrow$ *finish. KO* $\Rightarrow k := k + 1$, *back to 2.*

Context and questions

Approximate solution

- approximate solution $U^{k,i}$ does not solve $\mathcal{A}(U^{k,i}) = F$

Numerical method

- underlying numerical method: the vector $U^{k,i}$ is associated with a (piecewise polynomial) approximation $u_h^{k,i}$

Partial differential equation

- underlying PDE, u its weak solution: $A(u) = f$

Question (Stopping criteria)

- *What is a good stopping criterion for the nonlinear solver?*
- *What is a good stopping criterion for the linear solver?*

Question (Error)

- *How big is the error $\|u - u_h^{k,i}\|$ on Newton step k and algebraic solver step i, how is it distributed?*

Quasi-linear elliptic problem

Quasi-linear elliptic problem

$$\begin{aligned} -\nabla\cdot\boldsymbol{\sigma}(u,\nabla u) &= f && \text{in } \Omega,\\ u &= 0 && \text{on } \partial\Omega \end{aligned}$$

- quasi-linear diffusion problem
$$\boldsymbol{\sigma}(v,\boldsymbol{\xi}) = \underline{\mathbf{A}}(v)\boldsymbol{\xi} \qquad \forall (v,\boldsymbol{\xi}) \in \mathbb{R}\times\mathbb{R}^d$$
- Leray–Lions problem
$$\boldsymbol{\sigma}(v,\boldsymbol{\xi}) = \underline{\mathbf{A}}(\boldsymbol{\xi})\boldsymbol{\xi} \qquad \forall \boldsymbol{\xi} \in \mathbb{R}^d$$
- $p > 1$, $q := \frac{p}{p-1}$, $f \in L^q(\Omega)$

Example

p-Laplacian: Leray–Lions setting with $\underline{\mathbf{A}}(\boldsymbol{\xi}) = |\boldsymbol{\xi}|^{p-2}\underline{\mathbf{I}}$

Nonlinear operator $A : V := W_0^{1,p}(\Omega) \to V'$

$$\langle A(u), v\rangle_{V',V} := (\boldsymbol{\sigma}(u,\nabla u), \nabla v)$$

Weak formulation

Find $u \in V$ such that $\quad A(u) = f$ in V'

Approximate solution and error measure

Approximate solution

- $u_h^{k,i} \in V(\mathcal{T}_h) \not\subset V$, $u_h^{k,i}$ not necessarily in V
- $V(\mathcal{T}_h) := \{v \in L^p(\Omega),\ v|_K \in W^{1,p}(K) \quad \forall K \in \mathcal{T}_h\}$

Error measure

$$\mathcal{J}_u(u_h^{k,i}) := \sup_{\varphi\in V;\ \|\nabla\varphi\|_p=1} (\boldsymbol{\sigma}(u,\nabla u) - \boldsymbol{\sigma}(u_h^{k,i},\nabla u_h^{k,i}), \nabla\varphi) + \mathcal{J}_{u,\mathrm{NC}}(u_h^{k,i})$$

$$\mathcal{J}_{u,\mathrm{NC}}(u_h^{k,i}) := \left\{ \sum_{K\in\mathcal{T}_h} \sum_{e\in\mathcal{E}_K} h_e^{1-q} \|[\![u - u_h^{k,i}]\!]\|_{q,e}^q \right\}^{1/q}$$

- weak difference of the fluxes (dual norm of the residual) + nonconformity (computable jump term)
- there holds $\mathcal{J}_u(u_h^{k,i}) = 0$ if and only if $u = u_h^{k,i}$
- physical relevance: **strong** difference of the **fluxes** + **nonconformity**

$$\mathcal{J}_u(u_h^{k,i}) \le \mathcal{J}_u^{\mathrm{up}}(u_h^{k,i}) := \|\boldsymbol{\sigma}(u,\nabla u) - \boldsymbol{\sigma}(u_h^{k,i},\nabla u_h^{k,i})\|_q + \mathcal{J}_{u,\mathrm{NC}}(u_h^{k,i})$$

A posteriori error estimate

Assumption A (Total flux reconstruction)

There exists a flux reconstruction $\mathbf{t}_h^{k,i} \in \mathbf{H}^q(\mathrm{div},\Omega)$ and an algebraic remainder $\rho_h^{k,i} \in L^q(\Omega)$ such that

$$\nabla\cdot\mathbf{t}_h^{k,i} = f_h - \rho_h^{k,i},$$

with the data approximation f_h s.t. $(f_h,1)_K = (f,1)_K \ \forall K \in \mathcal{T}_h$.

Theorem (A guaranteed a posteriori error estimate)

Let

- *$u \in V$ be the weak solution,*
- *$u_h^{k,i} \in V(\mathcal{T}_h)$ be arbitrary,*
- *Assumption A hold.*

Then there holds

$$\mathcal{J}_u(u_h^{k,i}) \le \overline{\eta}^{k,i},$$

where $\overline{\eta}^{k,i}$ is fully computable from $u_h^{k,i}$, $\mathbf{t}_h^{k,i}$, and $\rho_h^{k,i}$.

Distinguishing error components

Assumption B (Discretization, linearization, and algebraic errors)

There exist fluxes $\mathbf{d}_h^{k,i}, \mathbf{l}_h^{k,i}, \mathbf{a}_h^{k,i} \in [L^q(\Omega)]^d$ such that

(i) $\mathbf{d}_h^{k,i} + \mathbf{l}_h^{k,i} + \mathbf{a}_h^{k,i} = \mathbf{t}_h^{k,i}$;

(ii) *as the linear solver converges, $\|\mathbf{a}_h^{k,i}\|_q \to 0$;*

(iii) *as the nonlinear solver converges, $\|\mathbf{l}_h^{k,i}\|_q \to 0$.*

Comments

- $\mathbf{d}_h^{k,i}$: *discretization flux reconstruction*
- $\mathbf{l}_h^{k,i}$: *linearization **error** flux reconstruction*
- $\mathbf{a}_h^{k,i}$: *algebraic **error** flux reconstruction*

Estimate distinguishing error components

Theorem (Estimate distinguishing different error components)

Let

- *$u \in V$ be the weak solution,*
- *$u_h^{k,i} \in V(\mathcal{T}_h)$ be arbitrary,*
- *Assumptions A and B hold.*

Then there holds

$$\mathcal{J}_u(u_h^{k,i}) \le \eta^{k,i} := \eta_{\mathrm{disc}}^{k,i} + \eta_{\mathrm{lin}}^{k,i} + \eta_{\mathrm{alg}}^{k,i} + \eta_{\mathrm{rem}}^{k,i} + \eta_{\mathrm{quad}}^{k,i} + \eta_{\mathrm{osc}}^{k,i}.$$

Estimators

- *discretization* estimator
$$\eta_{\mathrm{disc},K}^{k,i} := 2^{1/p}\left(\|\overline{\boldsymbol{\sigma}}_h^{k,i} + \mathbf{d}_h^{k,i}\|_{q,K} + \left\{\sum_{e\in\mathcal{E}_K} h_e^{1-q}\|[\![u_h^{k,i}]\!]\|_{q,e}^q\right\}^{\frac{1}{q}}\right)$$
- *linearization* estimator
$$\eta_{\mathrm{lin},K}^{k,i} := \|\mathbf{l}_h^{k,i}\|_{q,K}$$
- *algebraic* estimator
$$\eta_{\mathrm{alg},K}^{k,i} := \|\mathbf{a}_h^{k,i}\|_{q,K}$$
- *algebraic remainder estimator*
$$\eta_{\mathrm{rem},K}^{k,i} := h_\Omega\|\rho_h^{k,i}\|_{q,K}$$
- *quadrature estimator*
$$\eta_{\mathrm{quad},K}^{k,i} := \|\boldsymbol{\sigma}(u_h^{k,i},\nabla u_h^{k,i}) - \overline{\boldsymbol{\sigma}}_h^{k,i}\|_{q,K}$$
- *data oscillation estimator*
$$\eta_{\mathrm{osc},K}^{k,i} := C_{\mathrm{P},p}h_K\|f - f_h\|_{q,K}$$
- $\eta_{\cdot,K}^{k,i} := \left\{\sum_{K\in\mathcal{T}_h}(\eta_{\cdot,K}^{k,i})^q\right\}^{1/q}$

Stopping criteria

Global stopping criteria

- stop whenever:
$$\eta_{\mathrm{rem}}^{k,i} \le \gamma_{\mathrm{rem}}\max\{\eta_{\mathrm{disc}}^{k,i},\eta_{\mathrm{lin}}^{k,i},\eta_{\mathrm{alg}}^{k,i}\},$$
$$\eta_{\mathrm{alg}}^{k,i} \le \gamma_{\mathrm{alg}}\max\{\eta_{\mathrm{disc}}^{k,i},\eta_{\mathrm{lin}}^{k,i}\},$$
$$\eta_{\mathrm{lin}}^{k,i} \le \gamma_{\mathrm{lin}}\eta_{\mathrm{disc}}^{k,i}$$
- $\gamma_{\mathrm{rem}},\gamma_{\mathrm{alg}},\gamma_{\mathrm{lin}} \approx 0.1$

Local stopping criteria

- stop whenever:
$$\eta_{\mathrm{rem},K}^{k,i} \le \gamma_{\mathrm{rem},K}\max\{\eta_{\mathrm{disc},K}^{k,i},\eta_{\mathrm{lin},K}^{k,i},\eta_{\mathrm{alg},K}^{k,i}\} \qquad \forall K\in\mathcal{T}_h,$$
$$\eta_{\mathrm{alg},K}^{k,i} \le \gamma_{\mathrm{alg},K}\max\{\eta_{\mathrm{disc},K}^{k,i},\eta_{\mathrm{lin},K}^{k,i}\} \qquad \forall K\in\mathcal{T}_h,$$
$$\eta_{\mathrm{lin},K}^{k,i} \le \gamma_{\mathrm{lin},K}\eta_{\mathrm{disc},K}^{k,i} \qquad \forall K\in\mathcal{T}_h$$
- $\gamma_{\mathrm{rem},K},\gamma_{\mathrm{alg},K},\gamma_{\mathrm{lin},K} \approx 0.1$

Assumption for efficiency

Assumption C (Approximation property)

For all $K\in\mathcal{T}_h$, there holds

$$\|\overline{\boldsymbol{\sigma}}_h^{k,i} + \mathbf{d}_h^{k,i}\|_{q,K} \lesssim \eta_{\sharp,\mathfrak{T}_K}^{k,i} + \eta_{\mathrm{osc},\mathfrak{T}_K}^{k,i},$$

where

$$\eta_{\sharp,\mathfrak{T}_K}^{k,i} := \left\{\sum_{K'\in\mathfrak{T}_K} h_{K'}^q\|f_h + \nabla\cdot\overline{\boldsymbol{\sigma}}_h^{k,i}\|_{q,K'}^q + \sum_{e\in\mathfrak{E}_K^{\mathrm{int}}} h_e\|[\![\overline{\boldsymbol{\sigma}}_h^{k,i}\cdot\mathbf{n}_e]\!]\|_{q,e}^q + \sum_{e\in\mathcal{E}_K} h_e^{1-q}\|[\![u_h^{k,i}]\!]\|_{q,e}^q\right\}^{\frac{1}{q}}.$$

Global efficiency

Theorem (Global efficiency)

Let the mesh $\mathcal{T}_h$ be shape-regular and let the global stopping criteria hold. Recall that $\mathcal{J}_u(u_h^{k,i}) \le \eta^{k,i}$. Then, under Assumption C,

$$\eta^{k,i} \lesssim \mathcal{J}_u(u_h^{k,i}) + \eta_{\mathrm{quad}}^{k,i} + \eta_{\mathrm{osc}}^{k,i},$$

where $\lesssim$ means up to a constant independent of σ and q.

- **robustness** with respect to the **nonlinearity** thanks to the choice of the dual norm as error measure

Local efficiency

Theorem (Local efficiency)

Let the mesh $\mathcal{T}_h$ be shape-regular and let the local stopping criteria hold. Then, under Assumption C,

$$\eta_{\mathrm{disc},K}^{k,i} + \eta_{\mathrm{lin},K}^{k,i} + \eta_{\mathrm{alg},K}^{k,i} + \eta_{\mathrm{rem},K}^{k,i} \lesssim \mathcal{J}_{u,\mathfrak{T}_K}^{\mathrm{up}}(u_h^{k,i}) + \eta_{\mathrm{quad},\mathfrak{T}_K}^{k,i} + \eta_{\mathrm{osc},\mathfrak{T}_K}^{k,i}$$

for all $K\in\mathcal{T}_h$.

- robustness and **local efficiency** for an upper bound on the dual norm

Algebraic error flux reconstruction and algebraic remainder

Construction of $\mathbf{a}_h^{k,i}$ and $\rho_h^{k,i}$

- On linearization step k and algebraic step i, we have
$$\mathbb{A}^{k-1}U^{k,i} = F^{k-1} - R^{k,i}.$$
- Do ν additional steps of the algebraic solver, yielding
$$\mathbb{A}^{k-1}U^{k,i+\nu} = F^{k-1} - R^{k,i+\nu}.$$
- Construct the function $\rho_h^{k,i}$ from the algebraic residual vector $R^{k,i+\nu}$ (lifting into appropriate discrete space).
- Suppose we can obtain discretization and linearization flux reconstructions $\mathbf{d}_h^{k,i}, \mathbf{l}_h^{k,i}$ on each algebraic step. Then set
$$\mathbf{a}_h^{k,i} := (\mathbf{d}_h^{k,i+\nu} + \mathbf{l}_h^{k,i+\nu}) - (\mathbf{d}_h^{k,i} + \mathbf{l}_h^{k,i}).$$
- ν chosen adaptively so that $\eta_{\mathrm{rem},K}^{k,i}$ or $\eta_{\mathrm{rem}}^{k,i}$ are small enough.
- Independent of the algebraic solver.

Nonconforming finite elements for the p-Laplacian

Discretization
Find $u_h \in V_h$ such that

$$(\sigma(\nabla u_h), \nabla v_h) = (f_h, v_h) \qquad \forall v_h \in V_h.$$

- $\sigma(\nabla u_h) = |\nabla u_h|^{p-2}\nabla u_h$
- V_h the Crouzeix–Raviart space
- $f_h := \Pi_0 f$
- leads to the system of nonlinear algebraic equations

$$\mathcal{A}(U) = F$$

Linearization

Linearization
Find $u_h^k \in V_h$ such that

$$(\sigma^{k-1}(\nabla u_h^k), \nabla \psi_e) = (f_h, \psi_e) \qquad \forall e \in \mathcal{E}_h^{\mathrm{int}}.$$

- $u_h^0 \in V_h$ yields the initial vector U^0
- fixed-point linearization

$$\sigma^{k-1}(\xi) := |\nabla u_h^{k-1}|^{p-2}\xi$$

- Newton linearization

$$\sigma^{k-1}(\xi) := |\nabla u_h^{k-1}|^{p-2}\xi + (p-2)|\nabla u_h^{k-1}|^{p-4} (\nabla u_h^{k-1} \otimes \nabla u_h^{k-1})(\xi - \nabla u_h^{k-1})$$

- leads to the system of linear algebraic equations

$$\mathbb{A}^{k-1} U^k = F^{k-1}$$

Algebraic solution

Algebraic solution
Find $u_h^{k,i} \in V_h$ such that

$$(\sigma^{k-1}(\nabla u_h^{k,i}), \nabla \psi_e) = (f_h, \psi_e) - R_e^{k,i} \qquad \forall e \in \mathcal{E}_h^{\mathrm{int}}.$$

- algebraic residual vector $R^{k,i} = \{R_e^{k,i}\}_{e \in \mathcal{E}_h^{\mathrm{int}}}$
- discrete system

$$\mathbb{A}^{k-1} U^k = F^{k-1} - R^{k,i}$$

Flux reconstructions

Definition (Construction of $(\mathbf{d}_h^{k,i} + \mathbf{l}_h^{k,i})$)
For all $K \in \mathcal{T}_h$,

$$(\mathbf{d}_h^{k,i} + \mathbf{l}_h^{k,i})|_K := -\sigma^{k-1}(\nabla u_h^{k,i})|_K + \frac{f_h|_K}{d}(\mathbf{x} - \mathbf{x}_K) - \sum_{e \in \mathcal{E}_K} \frac{R_e^{k,i}}{d|D_e|}(\mathbf{x} - \mathbf{x}_K)|_{K_e},$$

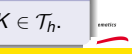
where, $R_e^{k,i} := (f_h, \psi_e) - (\sigma^{k-1}(\nabla u_h^{k,i}), \nabla \psi_e) \qquad \forall e \in \mathcal{E}_h^{\mathrm{int}}.$

Definition (Construction of $\mathbf{d}_h^{k,i}$)
For all $K \in \mathcal{T}_h$,

$$\mathbf{d}_h^{k,i}|_K := -\sigma(\nabla u_h^{k,i})|_K + \frac{f_h|_K}{d}(\mathbf{x} - \mathbf{x}_K) - \sum_{e \in \mathcal{E}_K} \frac{\bar{R}_e^{k,i}}{d|D_e|}(\mathbf{x} - \mathbf{x}_K)|_{K_e},$$

where $\bar{R}_e^{k,i} := (f_h, \psi_e) - (\sigma(\nabla u_h^{k,i}), \nabla \psi_e) \qquad \forall e \in \mathcal{E}_h^{\mathrm{int}}.$

Definition (Construction of $\overline{\sigma}_h^{k,i}$)
Set $\overline{\sigma}_h^{k,i} := \sigma(\nabla u_h^{k,i})$. Consequently, $\eta_{\mathrm{quad},K}^{k,i} = 0$ for all $K \in \mathcal{T}_h$.



Verification of the assumptions – upper bound

Lemma (Assumptions A and B)
Assumptions A and B hold.

Comments

- $\|\mathbf{a}_h^{k,i}\|_{q,K} \to 0$ as the linear solver converges by definition.
- $\|\mathbf{l}_h^{k,i}\|_{q,K} \to 0$ as the nonlinear solver converges by the construction of $\mathbf{l}_h^{k,i}$.
- Both $(\mathbf{d}_h^{k,i} + \mathbf{l}_h^{k,i})$ and $\mathbf{d}_h^{k,i}$ belong to $\mathbf{RTN}_0(\mathcal{S}_h) \Rightarrow$ $\mathbf{a}_h^{k,i} \in \mathbf{RTN}_0(\mathcal{S}_h)$ and $\mathbf{t}_h^{k,i} \in \mathbf{RTN}_0(\mathcal{S}_h)$.

Verification of the assumptions – efficiency

Lemma (Assumption C)
Assumption C holds.

Comments

- $\mathbf{d}_h^{k,i}$ close to $\sigma(\nabla u_h^{k,i})$
- approximation properties of Raviart–Thomas–Nédélec spaces

Discontinuous Galerkin for the quasi-linear diffusion

Discretization
Find $u_h \in V_h := \mathbb{P}_m(\mathcal{T}_h)$, $m \geq 1$, such that, for all $v_h \in V_h$,

$$(\sigma(u_h, \nabla u_h), \nabla v_h) - \sum_{e\in\mathcal{E}_h} \{\langle \{\!\{\sigma(u_h, \nabla u_h)\}\!\}\cdot\mathbf{n}_e, [\![v_h]\!]\rangle_e$$
$$+ \theta\langle \{\!\{\underline{\mathbf{A}}(u_h)\nabla v_h\}\!\}\cdot\mathbf{n}_e, [\![u_h]\!]\rangle_e\} + \sum_{e\in\mathcal{E}_h}\langle \bar{\alpha}_e h_e^{-1}[\![u_h]\!], [\![v_h]\!]\rangle_e = (f, v_h).$$

- $\theta \in \{-1, 0, 1\}$
- $\bar{\alpha}_e := \|\underline{\mathbf{A}}\|_{L^\infty(\mathbb{R})}\chi_e$, χ_e large enough
- leads to the system of nonlinear algebraic equations

$$\mathcal{A}(U) = F$$

Linearization

Linearization
Find $u_h^k \in V_h$ such that, for all $K \in \mathcal{T}_h$ and all $j \in \mathcal{C}_K := \{1, \dots, \dim(\mathbb{P}_m(K))\}$,

$$(\sigma^{k-1}(u_h^k, \nabla u_h^k), \nabla\psi_{K,j}) - \sum_{e\in\mathcal{E}_h} \{\langle \{\!\{\sigma^{k-1}(u_h^k, \nabla u_h^k)\}\!\}\cdot\mathbf{n}_e, [\![\psi_{K,j}]\!]\rangle_e$$
$$+\theta\langle \{\!\{\underline{\mathbf{A}}^{k-1}(u_h^k)\nabla\psi_{K,j}\}\!\}\cdot\mathbf{n}_e, [\![u_h^k]\!]\rangle_e\} + \sum_{e\in\mathcal{E}_h}\langle \bar{\alpha}_e h_e^{-1}[\![u_h^k]\!], [\![\psi_{K,j}]\!]\rangle_e = (f, \psi_{K,j}).$$

- $u_h^0 \in V_h$ yields the initial vector U^0
- fixed-point linearization $\sigma^{k-1}(v, \xi) := \underline{\mathbf{A}}(u_h^{k-1})\xi$
- Newton linearization

$$\sigma^{k-1}(v, \xi) := \underline{\mathbf{A}}(u_h^{k-1})\xi + (v - u_h^{k-1})\partial_v\underline{\mathbf{A}}(u_h^{k-1})\nabla u_h^{k-1},$$
$$\underline{\mathbf{A}}^{k-1}(v) := \underline{\mathbf{A}}(u_h^{k-1}) + \partial_v\underline{\mathbf{A}}(u_h^{k-1})(v - u_h^{k-1})$$

- leads to the system of linear algebraic equations

$$\mathbb{A}^{k-1}U^k = F^{k-1}$$

Algebraic solution

Algebraic solution
Find $u_h^{k,i} \in V_h$ such that

$$(\sigma^{k-1}(u_h^{k,i}, \nabla u_h^{k,i}), \nabla\psi_{K,j}) - \sum_{e\in\mathcal{E}_h} \{\langle \{\!\{\sigma^{k-1}(u_h^{k,i}, \nabla u_h^{k,i})\}\!\}\cdot\mathbf{n}_e, [\![\psi_{K,j}]\!]\rangle_e$$
$$+ \theta\langle \{\!\{\underline{\mathbf{A}}^{k-1}(u_h^{k,i})\nabla\psi_{K,j}\}\!\}\cdot\mathbf{n}_e, [\![u_h^{k,i}]\!]\rangle_e\} + \sum_{e\in\mathcal{E}_h}\langle \bar{\alpha}_e h_e^{-1}[\![u_h^{k,i}]\!], [\![\psi_{K,j}]\!]\rangle_e$$
$$= (f, \psi_{K,j}) - R_{K,j}^{k,i}.$$

- algebraic residual vector $R^{k,i} = \{R_{K,j}^{k,i}\}_{K\in\mathcal{T}_h, j\in\mathcal{C}_K}$
- discrete system

$$\mathbb{A}^{k-1}U^k = F^{k-1} - R^{k,i}$$

Flux reconstructions

Definition (Construction of $(\mathbf{d}_h^{k,i} + \mathbf{l}_h^{k,i}) \in \mathbf{RTN}_l(\mathcal{T}_h)$, $l := m-1/m$)
For all $K \in \mathcal{T}_h$ and all $e \in \mathcal{E}_K$,

$$\langle(\mathbf{d}_h^{k,i}+\mathbf{l}_h^{k,i})\cdot\mathbf{n}_e, q_h\rangle_e := \langle -\{\!\{\sigma^{k-1}(u_h^{k,i}, \nabla u_h^{k,i})\}\!\}\cdot\mathbf{n}_e + \bar{\alpha}_e h_e^{-1}[\![u_h^{k,i}]\!], q_h\rangle_e,$$
$$(\mathbf{d}_h^{k,i} + \mathbf{l}_h^{k,i}, \mathbf{r}_h)_K := -(\sigma^{k-1}(u_h^{k,i}, \nabla u_h^{k,i}), \mathbf{r}_h)_K$$
$$+ \theta\sum_{e\in\mathcal{E}_K} w_e\langle\underline{\mathbf{A}}^{k-1}(u_h^{k,i})\mathbf{r}_h\cdot\mathbf{n}_e, [\![u_h^{k,i}]\!]\rangle_e,$$

for all $q_h \in \mathbb{P}_l(e)$ and all $\mathbf{r}_h \in [\mathbb{P}_{l-1}(K)]^d$.

Definition (Construction of $\mathbf{d}_h^{k,i} \in \mathbf{RTN}_l(\mathcal{T}_h)$, $l := m-1$ or $l := m$)
For all $K \in \mathcal{T}_h$ and all $e \in \mathcal{E}_K$,

$$\langle\mathbf{d}_h^{k,i}\cdot\mathbf{n}_e, q_h\rangle_e := \langle -\{\!\{\sigma(u_h^{k,i}, \nabla u_h^{k,i})\}\!\}\cdot\mathbf{n}_e + \bar{\alpha}_e h_e^{-1}[\![u_h^{k,i}]\!], q_h\rangle_e,$$
$$(\mathbf{d}_h^{k,i}, \mathbf{r}_h)_K := -(\sigma(u_h^{k,i}, \nabla u_h^{k,i}), \mathbf{r}_h)_K + \theta\sum_{e\in\mathcal{E}_K} w_e\langle\underline{\mathbf{A}}(u_h^{k,i})\mathbf{r}_h\cdot\mathbf{n}_e, [\![u_h^{k,i}]\!]\rangle_e,$$

for all $q_h \in \mathbb{P}_l(e)$ and all $\mathbf{r}_h \in [\mathbb{P}_{l-1}(K)]^d$.

Verification of the assumptions – upper bound

Definition (Construction of f_h, $\overline{\sigma}_h^{k,i}$)
Set $f_h := \Pi_l f$ and $\overline{\sigma}_h^{k,i} := \mathbf{l}_l^{\mathrm{RTN}}(\sigma(u_h^{k,i}, \nabla u_h^{k,i}))$.

Lemma (Assumptions A and B)
Assumptions A and B hold.

Comments

- $\|\mathbf{a}_h^{k,i}\|_{q,K} \to 0$ as the linear solver converges by definition.
- $\|\mathbf{l}_h^{k,i}\|_{q,K} \to 0$ as the nonlinear solver converges by the construction of $\mathbf{l}_h^{k,i}$.
- Both $(\mathbf{d}_h^{k,i} + \mathbf{l}_h^{k,i})$ and $\mathbf{d}_h^{k,i}$ belong to $\mathbf{RTN}_l(\mathcal{T}_h) \Rightarrow$ $\mathbf{a}_h^{k,i} \in \mathbf{RTN}_l(\mathcal{T}_h)$ and $\mathbf{t}_h^{k,i} \in \mathbf{RTN}_l(\mathcal{T}_h)$.

Verification of the assumptions – efficiency

Lemma (Assumption C)
Assumption C holds.

Comments

- $\mathbf{d}_h^{k,i}$ close to $\overline{\sigma}_h^{k,i}$
- approximation properties of Raviart–Thomas–Nédélec spaces

Summary

Discretization methods

- conforming finite elements
- nonconforming finite elements
- discontinuous Galerkin
- various finite volumes
- mixed finite elements

Linearizations

- fixed point
- Newton

Linear solvers

- independent of the linear solver

. . . all Assumptions A to C verified

Numerical experiment I

Model problem

- p-Laplacian

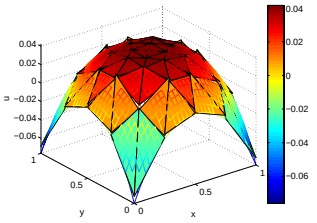
$$\begin{aligned} \nabla\cdot(|\nabla u|^{p-2}\nabla u) &= f \quad \text{in } \Omega, \\ u &= u_0 \quad \text{on } \partial\Omega \end{aligned}$$

- weak solution (used to impose the Dirichlet BC)

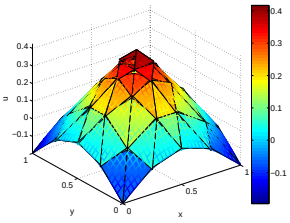
$$u(x,y) = -\tfrac{p-1}{p}\left((x-\tfrac{1}{2})^2+(y-\tfrac{1}{2})^2\right)^{\frac{p}{2(p-1)}} + \tfrac{p-1}{p}\left(\tfrac{1}{2}\right)^{\frac{p}{p-1}}$$

- tested values $p = 1.5$ and 10
- nonconforming finite elements

Analytical and approximate solutions

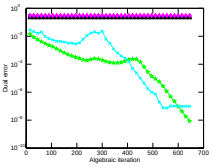


Case $p = 1.5$

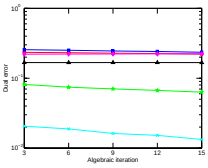


Case $p = 10$

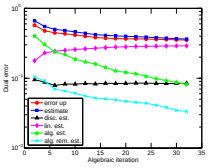
Error and estimators as a function of CG iterations, $p = 10$, 6th level mesh, 6th Newton step.



Newton

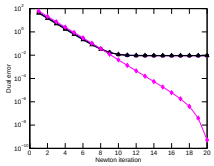


inexact Newton

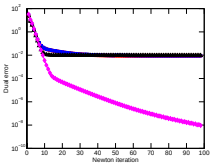


ad. inexact Newton

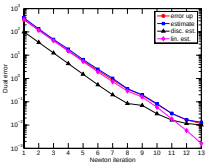
Error and estimators as a function of Newton iterations, $p = 10$, 6th level mesh



Newton

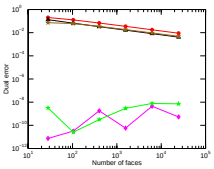


inexact Newton

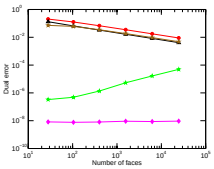


ad. inexact Newton

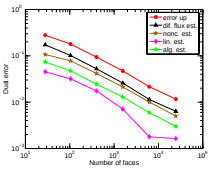
Error and estimators, $p = 10$



Newton

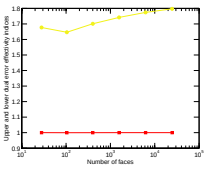


inexact Newton

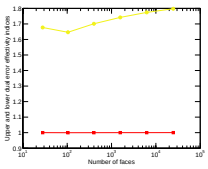


ad. inexact Newton

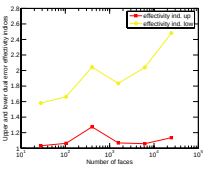
Effectivity indices, $p = 10$



Newton

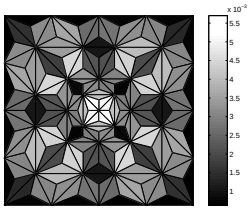


inexact Newton

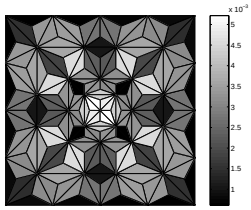


ad. inexact Newton

Error distribution, $p = 10$

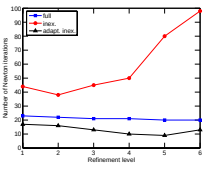


Estimated error distribution

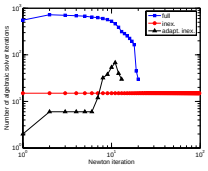


Exact error distribution

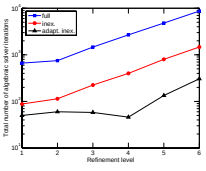
Newton and algebraic iterations, $p = 10$



Newton it. / refinement

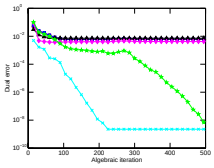


alg. it. / Newton step

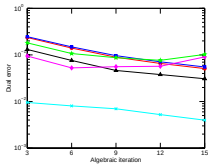


alg. it. / refinement

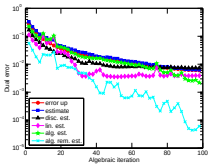
Error and estimators as a function of CG iterations, $p = 1.5$, 6th level mesh, 1st Newton step.



Newton

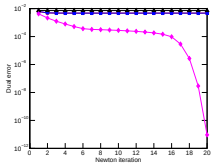


inexact Newton

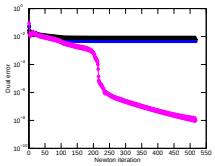


ad. inexact Newton

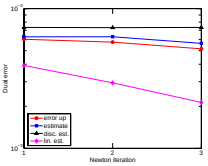
Error and estimators as a function of Newton iterations, $p = 1.5$, 6th level mesh



Newton

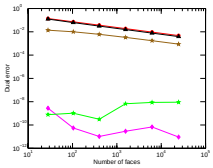


inexact Newton

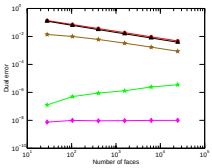


ad. inexact Newton

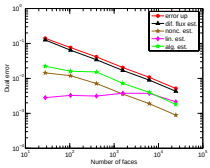
Error and estimators, $p = 1.5$



Newton

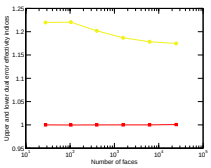


inexact Newton

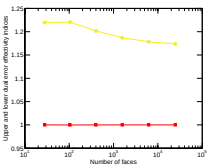


ad. inexact Newton

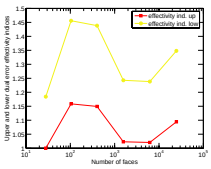
Effectivity indices, $p = 1.5$



Newton

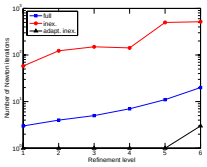


inexact Newton

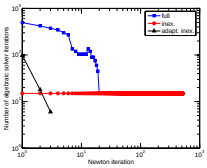


ad. inexact Newton

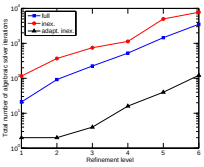
Newton and algebraic iterations, $p = 1.5$



Newton it. / refinement



alg. it. / Newton step



alg. it. / refinement

Numerical experiment II

Model problem

- p-Laplacian

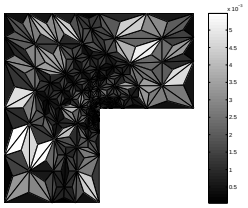
$$\nabla\cdot(|\nabla u|^{p-2}\nabla u) = f \quad \text{in } \Omega,$$
$$u = u_0 \quad \text{on } \partial\Omega$$

- weak solution (used to impose the Dirichlet BC)

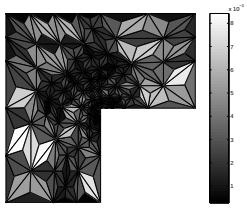
$$u(r,\theta) = r^{\frac{7}{8}}\sin(\theta\tfrac{7}{8})$$

- $p = 4$, L-shape domain, singularity in the origin (Carstensen and Klose (2003))
- nonconforming finite elements

Error distribution on an adaptively refined mesh

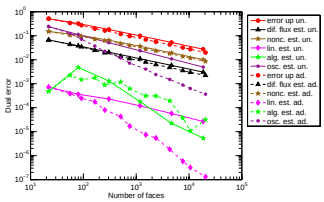


Estimated error distribution

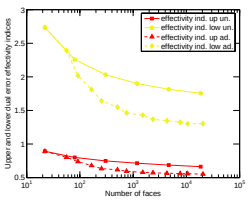


Exact error distribution

Estimated and actual errors and the effectivity index

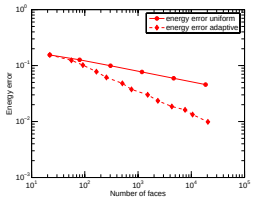


Estimated and actual errors

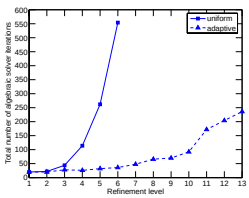


Effectivity index

Energy error and overall performance



Energy error



Overall performance

Two-phase flow in porous media

Two-phase flow in porous media

$$\begin{aligned} \partial_t(\phi s_\alpha) + \nabla\cdot\mathbf{u}_\alpha &= q_\alpha, && \alpha\in\{\mathrm{n},\mathrm{w}\},\\ -\lambda_\alpha(s_\mathrm{w})\underline{\mathbf{K}}(\nabla p_\alpha + \rho_\alpha g\nabla z) &= \mathbf{u}_\alpha, && \alpha\in\{\mathrm{n},\mathrm{w}\},\\ s_\mathrm{n}+s_\mathrm{w} &= 1,\\ p_\mathrm{n}-p_\mathrm{w} &= p_c(s_\mathrm{w}) \end{aligned}$$

Mathematical issues

- coupled system
- unsteady, nonlinear
- elliptic–parabolic degenerate type
- dominant advection

Two-phase flow in porous media

Theorem (A posteriori error estimate distinguishing the error components)

Let

- n *be the time step,*
- k *be the linearization step,*
- i *be the algebraic solver step,*

with the approximations $(s_{\mathrm{w},h\tau}^{n,k,i}, p_{\mathrm{w},h\tau}^{n,k,i})$. *Then*

$$|||(s_\mathrm{w}-s_{\mathrm{w},h\tau}^{n,k,i}, p_\mathrm{w}-p_{\mathrm{w},h\tau}^{n,k,i})|||_{I_n} \le \eta_\mathrm{sp}^{n,k,i}+\eta_\mathrm{tm}^{n,k,i}+\eta_\mathrm{lin}^{n,k,i}+\eta_\mathrm{alg}^{n,k,i}.$$

Error components

- $\eta_\mathrm{sp}^{n,k,i}$: spatial discretization
- $\eta_\mathrm{tm}^{n,k,i}$: temporal discretization
- $\eta_\mathrm{lin}^{n,k,i}$: linearization
- $\eta_\mathrm{alg}^{n,k,i}$: algebraic solver

Local estimators

- *spatial estimators*

$$\begin{aligned} \eta_{\mathrm{sp},K}^{n,k,i}(t) := \Bigg\{ &\sum_{\alpha\in\{\mathrm{n},\mathrm{w}\}} \big(\|\mathbf{d}_{\alpha,h}^{n,k,i}-\mathbf{v}_\alpha(p_{\mathrm{w},h}^{n,k,i}, s_{\mathrm{w},h}^{n,k,i})\|_K\\ &+ h_K/\pi\|q_\alpha^n-\partial_t^n(\phi s_{\alpha,h\tau}^{n,k,i})-\nabla\cdot\mathbf{u}_{\alpha,h}^{n,k,i}\|_K\big)^2\\ &+\big(\|\underline{\mathbf{K}}(\lambda_\mathrm{w}(s_{\mathrm{w},h\tau}^{n,k,i})+\lambda_\mathrm{n}(s_{\mathrm{w},h\tau}^{n,k,i}))\nabla(\mathfrak{p}(p_{\mathrm{w},h\tau}^{n,k,i}, s_{\mathrm{w},h\tau}^{n,k,i})-\bar{\mathfrak{p}}_{h\tau}^{n,k,i})\|_K(t)\big)^2\\ &+\big(\|\underline{\mathbf{K}}\nabla(\mathfrak{q}(s_{\mathrm{w},h\tau}^{n,k,i})-\bar{\mathfrak{q}}_{h\tau}^{n,k,i})\|_K(t)\big)^2\Bigg\}^{\frac12} \end{aligned}$$

- *temporal estimators*

$$\eta_{\mathrm{tm},K,\alpha}^{n,k,i}(t) := \|\mathbf{v}_\alpha(p_{\mathrm{w},h\tau}^{n,k,i}, s_{\mathrm{w},h\tau}^{n,k,i})(t)-\mathbf{v}_\alpha(p_{\mathrm{w},h\tau}^{n,k,i}, s_{\mathrm{w},h\tau}^{n,k,i})(t^n)\|_K \quad \alpha\in\{\mathrm{n},\mathrm{w}\}$$

- *linearization estimators*

$$\eta_{\mathrm{lin},K,\alpha}^{n,k,i} := \|\mathbf{l}_{\alpha,h}^{n,k,i}\|_K \qquad \alpha\in\{\mathrm{n},\mathrm{w}\}$$

- *algebraic estimators*

$$\eta_{\mathrm{alg},K,\alpha}^{n,k,i} := \|\mathbf{a}_{\alpha,h}^{n,k,i}\|_K \qquad \alpha\in\{\mathrm{n},\mathrm{w}\}$$

Global estimators

Global estimators

$$\eta_\mathrm{sp}^{n,k,i} := \left\{3\int_{I_n}\sum_{K\in\mathcal{T}_h^n}(\eta_{\mathrm{sp},K}^{n,k,i}(t))^2\,\mathrm{d}t\right\}^{\frac12},$$

$$\eta_\mathrm{tm}^{n,k,i} := \left\{\sum_{\alpha\in\{\mathrm{n},\mathrm{w}\}}\int_{I_n}\sum_{K\in\mathcal{T}_h^n}(\eta_{\mathrm{tm},K,\alpha}^{n,k,i}(t))^2\,\mathrm{d}t\right\}^{\frac12},$$

$$\eta_\mathrm{lin}^{n,k,i} := \left\{\sum_{\alpha\in\{\mathrm{n},\mathrm{w}\}}\tau^n\sum_{K\in\mathcal{T}_h^n}(\eta_{\mathrm{lin},K,\alpha}^{n,k,i})^2\right\}^{\frac12},$$

$$\eta_\mathrm{alg}^{n,k,i} := \left\{\sum_{\alpha\in\{\mathrm{n},\mathrm{w}\}}\tau^n\sum_{K\in\mathcal{T}_h^n}(\eta_{\mathrm{alg},K,\alpha}^{n,k,i})^2\right\}^{\frac12}$$

Cell-centered finite volume scheme

Cell-centered finite volume scheme

For all $1\le n\le N$, look for $s_{\mathrm{w},h}^n, \bar{p}_{\mathrm{w},h}^n$ such that

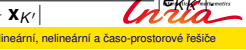
$$\phi\frac{s_{\mathrm{w},K}^n-s_{\mathrm{w},K}^{n-1}}{\tau^n}|K| + \sum_{e_{KK'}\in\mathcal{E}_K^\mathrm{int}} F_{\mathrm{w},e_{KK'}}(s_{\mathrm{w},h}^n, \bar{p}_{\mathrm{w},h}^n) = 0,$$

$$-\phi\frac{s_{\mathrm{w},K}^n-s_{\mathrm{w},K}^{n-1}}{\tau^n}|K| + \sum_{e_{KK'}\in\mathcal{E}_K^\mathrm{int}} F_{\mathrm{n},e_{KK'}}(s_{\mathrm{w},h}^n, \bar{p}_{\mathrm{w},h}^n) = 0,$$

where the fluxes are given by

$$F_{\mathrm{w},e_{KK'}}(s_{\mathrm{w},h}^n, \bar{p}_{\mathrm{w},h}^n) := -\frac{\lambda_\mathrm{w}(s_{\mathrm{w},K}^n)+\lambda_\mathrm{w}(s_{\mathrm{w},K'}^n)}{2}|\underline{\mathbf{K}}|\frac{\bar{p}_{\mathrm{w},K'}^n-\bar{p}_{\mathrm{w},K}^n}{|\mathbf{x}_K-\mathbf{x}_{K'}|}|e_{KK'}|,$$

$$F_{\mathrm{n},e_{KK'}}(s_{\mathrm{w},h}^n, \bar{p}_{\mathrm{w},h}^n) := -\frac{\lambda_\mathrm{n}(s_{\mathrm{w},K}^n)+\lambda_\mathrm{n}(s_{\mathrm{w},K'}^n)}{2}|\underline{\mathbf{K}}| \times\frac{\bar{p}_{\mathrm{w},K'}^n+\pi(s_{\mathrm{w},K'}^n)-(\bar{p}_{\mathrm{w},K}^n+\pi(s_{\mathrm{w},K}^n))}{|\mathbf{x}_K-\mathbf{x}_{K'}|}|e_{KK'}|$$



Linearization and algebraic solution

Linearization step k and algebraic step i

Couple $s_{\mathrm{w},h}^{n,k,i}, \bar{p}_{\mathrm{w},h}^{n,k,i}$ such that

$$\phi\frac{s_{\mathrm{w},K}^{n,k,i}-s_{\mathrm{w},K}^{n-1}}{\tau^n}|K| + \sum_{e_{KK'}\in\mathcal{E}_K^\mathrm{int}} F_{\mathrm{w},e_{KK'}}^{k-1}(s_{\mathrm{w},h}^{n,k,i}, \bar{p}_{\mathrm{w},h}^{n,k,i}) = -R_{\mathrm{w},K}^{n,k,i},$$

$$-\phi\frac{s_{\mathrm{w},K}^{n,k,i}-s_{\mathrm{w},K}^{n-1}}{\tau^n}|K| + \sum_{e_{KK'}\in\mathcal{E}_K^\mathrm{int}} F_{\mathrm{n},e_{KK'}}^{k-1}(s_{\mathrm{w},h}^{n,k,i}, \bar{p}_{\mathrm{w},h}^{n,k,i}) = -R_{\mathrm{n},K}^{n,k,i},$$

where the linearized fluxes are given by

$$\begin{aligned} F_{\alpha,e_{KK'}}^{k-1}(s_{\mathrm{w},h}^{n,k,i}, \bar{p}_{\mathrm{w},h}^{n,k,i}) :=& F_{\alpha,e_{KK'}}(s_{\mathrm{w},h}^{n,k-1}, \bar{p}_{\mathrm{w},h}^{n,k-1})\\ &+\sum_{M\in\{K,K'\}}\frac{\partial F_{\alpha,e_{KK'}}}{\partial s_{\mathrm{w},M}}(s_{\mathrm{w},h}^{n,k-1}, \bar{p}_{\mathrm{w},h}^{n,k-1})\cdot(s_{\mathrm{w},M}^{n,k,i}-s_{\mathrm{w},M}^{n,k-1})\\ &+\sum_{M\in\{K,K'\}}\frac{\partial F_{\alpha,e_{KK'}}}{\partial \bar{p}_{\mathrm{w},M}}(s_{\mathrm{w},h}^{n,k-1}, \bar{p}_{\mathrm{w},h}^{n,k-1})\cdot(\bar{p}_{\mathrm{w},M}^{n,k,i}-\bar{p}_{\mathrm{w},M}^{n,k-1}). \end{aligned}$$

Fluxes reconstructions and pressure postprocessing

Fluxes reconstructions

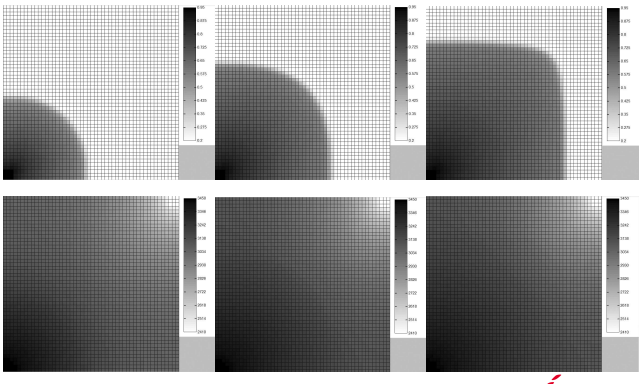
$$(\mathbf{d}_{\alpha,h}^{n,k,i}\cdot\mathbf{n}_K,1)_{e_{KK'}} := F_{\alpha,e_{KK'}}(s_{\mathrm{w},h}^{n,k,i},\bar{p}_{\mathrm{w},h}^{n,k,i}),$$
$$((\mathbf{d}_{\alpha,h}^{n,k,i}+\mathbf{l}_{\alpha,h}^{n,k,i})\cdot\mathbf{n}_K,1)_{e_{KK'}} := F_{\alpha,e_{KK'}}^{k-1}(s_{\mathrm{w},h}^{n,k,i},\bar{p}_{\mathrm{w},h}^{n,k,i}),$$
$$\mathbf{a}_{\alpha,h}^{n,k,i} := \mathbf{d}_{\alpha,h}^{n,k,i+\nu}+\mathbf{l}_{\alpha,h}^{n,k,i+\nu}-(\mathbf{d}_{\alpha,h}^{n,k,i}+\mathbf{l}_{\alpha,h}^{n,k,i})$$

Phase pressures postprocessing

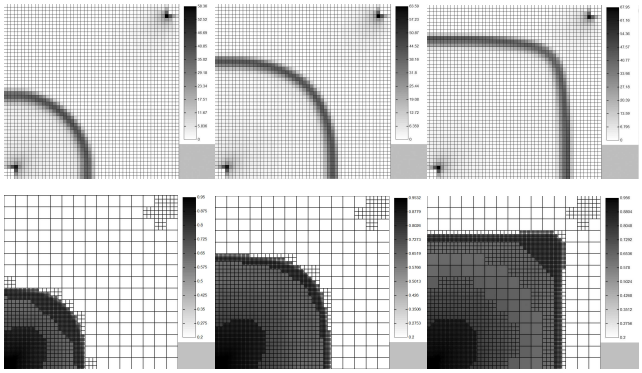
- Piecewise constant $\bar{p}_{\alpha,h}^{n,k,i}$ postprocessed to piecewise quadratic $p_{\alpha,h}^{n,k,i}$:

$$-\lambda_{\mathrm{w}}(s_{\mathrm{w},K}^{n,k,i})\underline{\mathbf{K}}\nabla(p_{\mathrm{w},h}^{n,k,i}|_K)=\mathbf{d}_{\mathrm{w},h}^{n,k,i}|_K,$$
$$p_{\mathrm{w},h}^{n,k,i}(\mathbf{x}_K)=\bar{p}_{\mathrm{w},K}^{n,k,i},$$
$$-\lambda_{\mathrm{n}}(s_{\mathrm{w},K}^{n,k,i})\underline{\mathbf{K}}\nabla(p_{\mathrm{n},h}^{n,k,i}|_K)=\mathbf{d}_{\mathrm{n},h}^{n,k,i}|_K,$$
$$p_{\mathrm{n},h}^{n,k,i}(\mathbf{x}_K)=\pi(s_{\mathrm{w},K}^{n,k,i})+\bar{p}_{\mathrm{w},K}^{n,k,i}$$

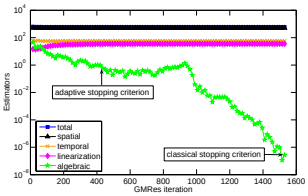
Water saturation/water pressure evolution



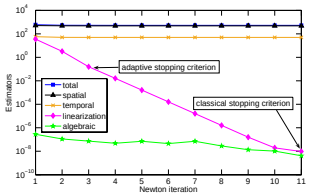
Estimators/meshes evolution



Estimators and stopping criteria

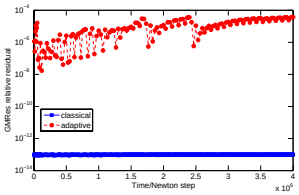


Estimators in function of GMRes iterations

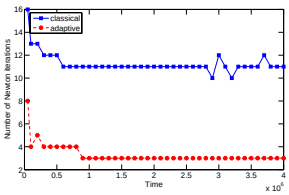


Estimators in function of Newton iterations

GMRes relative residual/Newton iterations

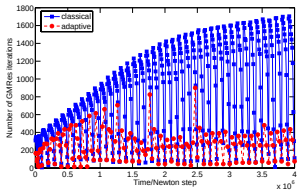


GMRes relative residual

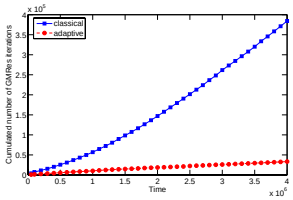


Newton iterations

GMRes iterations



Per time and Newton step



Cumulated

Conclusions

Entire adaptivity

- only a **necessary number** of **algebraic solver iterations** on each linearization step
- only a **necessary number** of **linearization iterations**
- **"smart online decisions"**: algebraic step / linearization step / space mesh refinement / time step modification
- important **computational savings**
- guaranteed and robust error upper bound via **a posteriori estimates**

Future directions

- other coupled nonlinear systems
- convergence and optimality

Bibliography

Bibliography

- VOHRALÍK M., *A posteriori error estimates for efficiency and error control in numerical simulations*, Lecture notes, Course NM497, Université Pierre et Marie Curie, Paris, 2012.
- JIRÁNEK P., STRAKOŠ Z., VOHRALÍK M., A posteriori error estimates including algebraic error and stopping criteria for iterative solvers. *SIAM J.Sci.Comput.* **32** (2010),1567–1590.
- ERN A., VOHRALÍK M., Adaptive inexact Newton methods with a posteriori stopping criteria for nonlinear diffusion PDEs, HAL Preprint 00681422.
- VOHRALÍK M., WHEELER M. F., A posteriori error estimates, stopping criteria, and adaptivity for two-phase flows, HAL Preprint 00633594.

Díky za Vaši pozornost!

Title: Seminar on Numerical Analysis & Winter School

Proceedings of the conference SNA'13
Rožnov pod Radhoštěm
January 21 – 25, 2013

Editors: Radim Blaheta, Jiří Starý, Hana Bílková

Published by: Institute of Geonics AS CR, Ostrava

Printed by: Moravapress s.r.o., Ostrava

First edition

Ostrava, 2013

copies 90

ISBN 978-80-86407-34-0